(continued from Part 3)

Further gates using MOS transistors

In common with most digital sub-systems, the keyboard encoder described previously needs AND gates and OR gates.

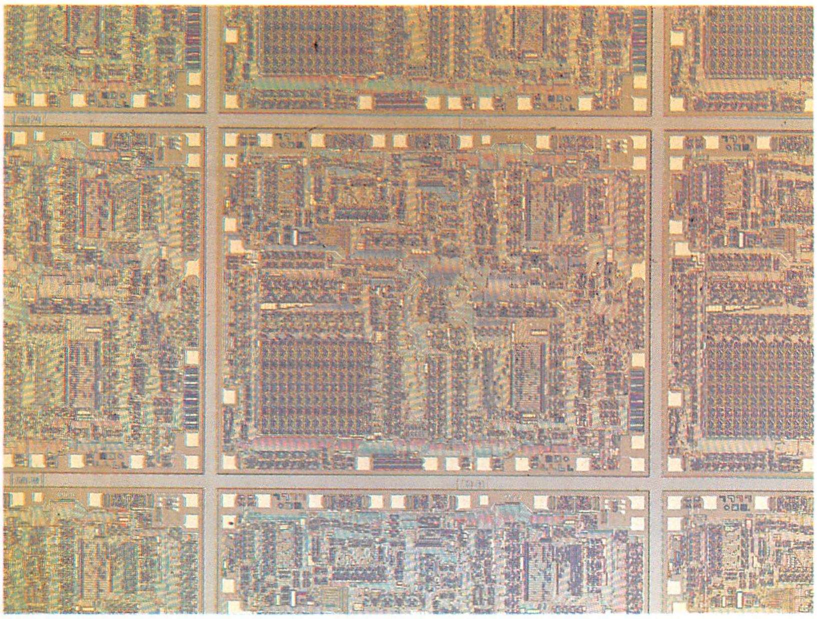
However, AND and OR gates are not as easily built with MOS transistors as certain other gate circuits. So, in practice, combinations of these alternative circuits are used to replace the AND and OR functions.

The first of these is shown in *figure 18*. The left side of the diagram shows that this circuit is an inverter (similar to the one shown in *figure 17*) but this time it has *two* input transistors in series. Now the output can only be connected to ground when both input transistors are on. (If more inputs are needed more input transistors can be connected in series). Another way of describing how this circuit works is to say that for the output to be low, all the inputs have to be high. From this it follows that if just one input is low then the output is high.

The use of 'H' and 'L' in function tables

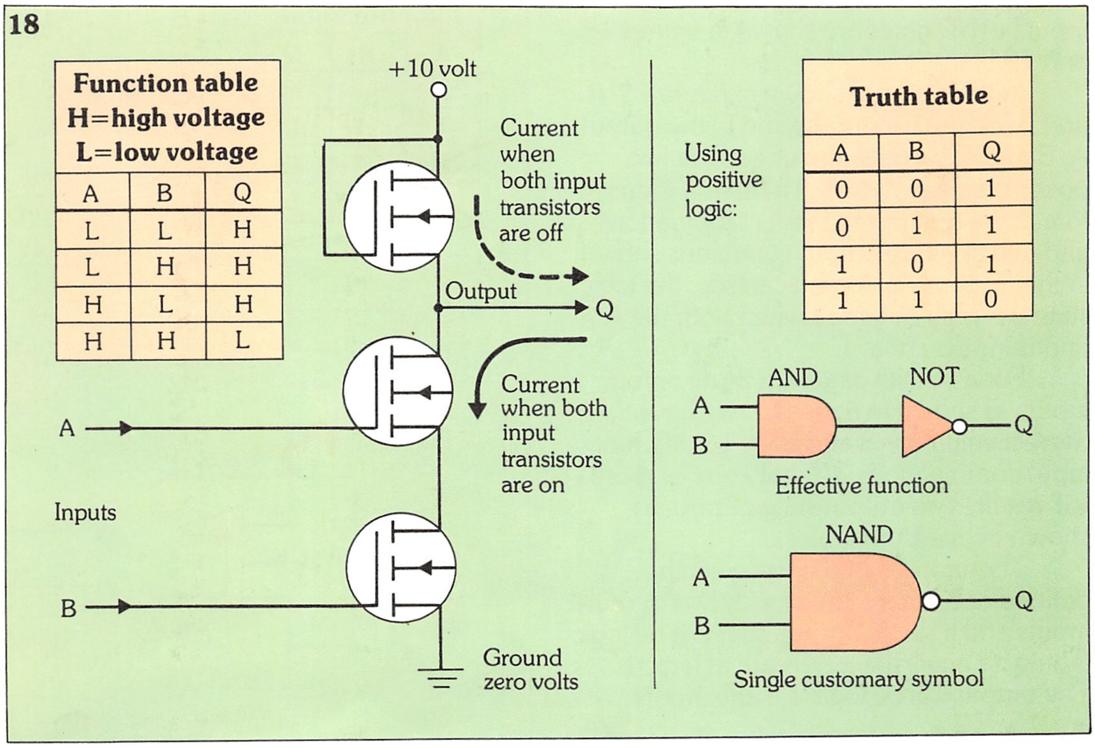
The way this type of gate works is shown in the function table on the left-hand side of *figure 18*, where 'H' and 'L' are used to indicate high and low voltage states. Assuming the same values we have used previously, H is associated with a potential of 10 volts and L with 0 volts or earth potential.

These symbols are commonly used in all the function tables supplied by the manufacturers of digital circuits. The reason for this is that we can't use the



An example of a CMOS integrated circuit. This particular chip is used in telephone equipment to handle keypad dialling. (Photo courtesy SGS).

18. MOS two-input positive NAND gate (left) and symbols for NAND function.



Function table
H=high voltage
L=low voltage

A	B	Q
L	L	H
L	H	H
H	L	H
H	H	L

Truth table

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

terms 'on' and 'off' since the outputs are always 'on' in a sense, as they are connected to the earth or the supply voltage. What distinguishes one electric state from another is the voltage level present in a wire. So H (high) and L (low) mean the same thing in real digital circuits as on and off in the imaginary switch circuits we have been using as examples.

NAND circuits

If **positive logic** is used and 1 replaces each H in the function table while 0 replaces each L, then the function table becomes a truth table as shown on the right in *figure 18*. This particular type of digital function is called NAND because it is the inverse of an AND gate. In other words it is a NOT-AND gate.

So, diagrammatically, you can represent the circuit by an AND gate followed by an inverter. The usual symbol is a combination of the two i.e. an AND gate with a little circle at the output indicating inversion. Since the circuit shown on the left of *figure 18* carries out NAND functions when positive logic is used, it is known as a positive NAND gate.

Truth tables of the most common NAND gates

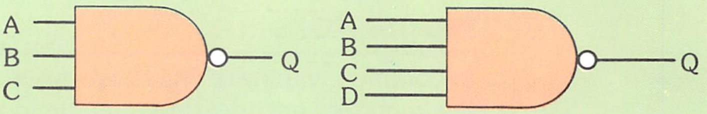
The truth tables for two, three and four input NAND gates are shown in *figures 18* and *19*.

In the diagram shown in *figure 18*, A and B are the two inputs and Q the output. A, B and Q can assume one of the two possible values: 0 or 1. There are, therefore, $2^2 = 4$ different input combinations and two possible output conditions shown in the truth table. As you can see, the logic state 0 is only obtained when both the A and B inputs are at 1.

For a NAND gate with three or four inputs as shown in *figure 19*, we have respectively $2^3 = 8$ and $2^4 = 16$ different input combinations, yet only one of these will result in an output state of zero, as shown by the truth tables.

Therefore, a NAND gate can be defined as a binary circuit with two or more inputs and a single output, that will be logic 0 only when all the inputs are at logic 1. The output will be logic 1 if any inputs are 0.

19



Truth table

A	B	C	Q
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Truth table

A	B	C	D	Q
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

20

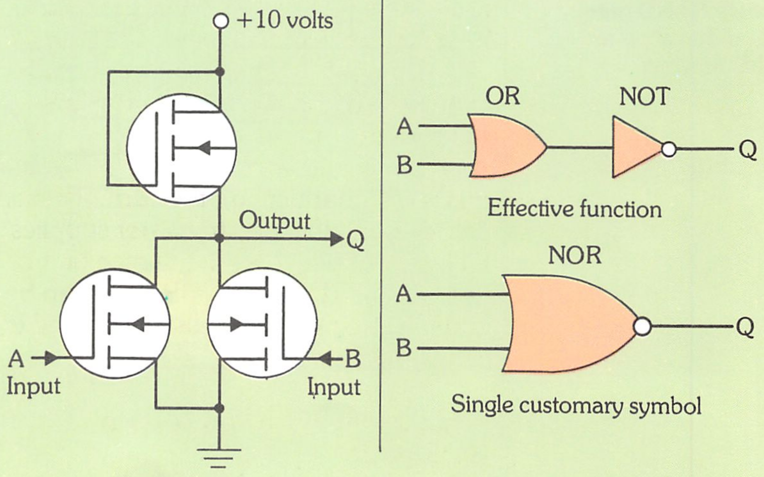
Function table

A	B	Q
L	L	H
L	H	L
H	L	L
H	H	L

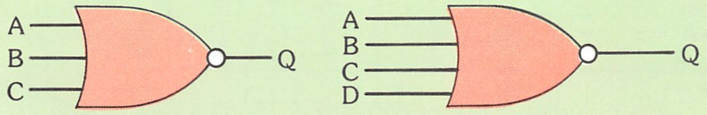
Using positive logic:

Truth table

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



21



Truth table

C	B	A	Q
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Truth table

D	C	B	A	Q
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

19. Three and four-input positive logic NAND gates, with truth tables.

20. MOS two-input positive NOR gate circuit (left) and symbols for NOR function.

21. Three or four-input positive logic NOR gates, with truth tables.

What is a NOR gate?

Figure 20 shows another type of MOS gate: in this case, a **positive NOR gate.** The NOR (NOT-OR) function is shown by the truth table on the right: its output equals 1 only when all the inputs are at 0. A NOR gate functions therefore like an OR circuit followed by an inverter, so the symbol commonly used to represent it is that of an OR gate followed by a small circle to indicate inversion. It is easy to see how the circuit on the left is a positive NOR gate: the application of a high signal to either one of the two input transistors switches the output to ground.

But if neither of the two input transistors is on (conducting), the current which passes along the load transistor supplies a high output signal. As in the case of the positive NAND gates, NOR gates can be made with as many inputs as required by simply adding input transistors.

Truth tables for the most common NOR gates

The symbol for a NOR gate with two inputs A and B, and an output Q, is shown in *figure 20*. Since the inputs can again have four distinct combinations ($2^2 = 4$), there are four distinct input/output conditions shown in the truth table. The output is in logic state 1 only when the inputs (A and B) are both 0.

Figure 21 shows the symbols and the truth tables for three and four input NOR gates. As before, the logic state 1 is present at the output only when all the inputs are at logic level 0. Therefore NOR can be defined as a binary circuit with two or more inputs and a single output which is at logic level 1 only when all the inputs are at 0.

How do we use MOS gates in the encoder?

In designing integrated digital circuits, NAND and NOR gates are preferred because they are easy and practical to make.

In *figure 22* you can finally see how the keyboard encoder is constructed. Compared with the original encoder in *figure 10*, **positive NAND circuits** take the place of all the gates and one inverter has been added.

However, the arrangement and the connection of the various components remain the same as before. Suppose that the number 5 key is pressed. Numerical line 5 will then be the only one to assume logic level 0 (since only its relative NAND gate has both inputs at 1).

The only NAND gates to receive input signals from the number 5 NAND gate are those relating to the output lines marked by numbers 1 and 4. This puts these lines to logic level 1 (remember a NAND gate gives a 1 output when at least one of its inputs is at logic level 0). The final result is the configuration 0101 in the output lines, which represents number 5 in binary code.

If we now look at the NAND gate producing the 'number ready' signal we can see that the inverter turns the 1 in input N into a 0 going into the NAND gate. Since both inputs to this gate are not 1, the output is 1.

This then is a very convenient way of building an encoder. All the logic gates are NAND gates apart from one inverter, and even this could be replaced by a two-input NAND gate with one input held permanently at logic level 1.

Positive and negative logic

We have been referring to positive logic NAND and NOR. It's important to understand *why* you need to know which type of logic is being used.

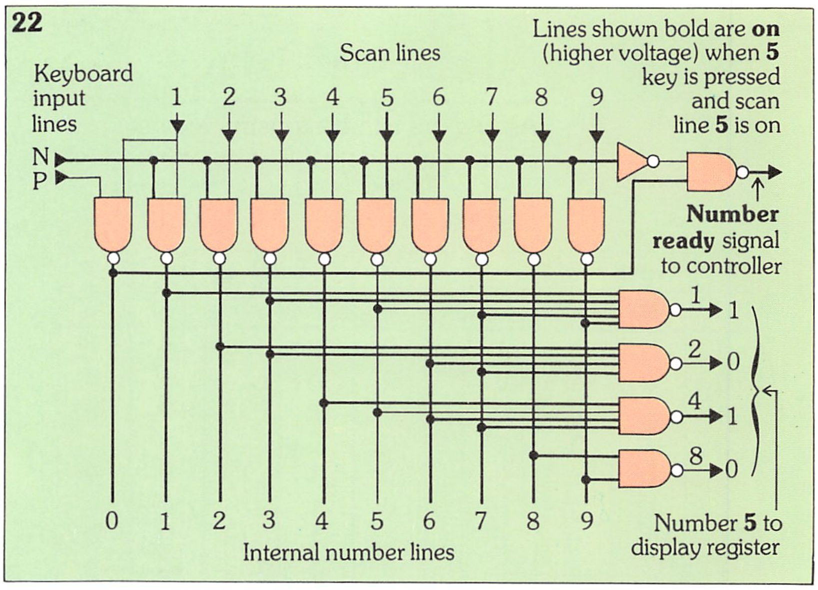
Let's take just the two NAND gates connected to scan lines 8 and 9 and the NAND gate connected to number 8 of the binary number output as in *figure 23*. Remember each NAND gate can be thought of as an AND gate followed by an inverter. If a dotted line is drawn around all three inverters it will also enclose the third NAND gate. In this box **negative logic** applies because L = YES and H = NO. A low from either of the inverters means yes one of the two keys has been pressed and the resulting low from the AND gate means yes, either key 8 or key 9 has been pressed. The function table for the AND gate is shown in *table 1*. Because negative logic is being used a low now equals 1 as shown in *table 2*. This truth table is the same as that for an **OR function**.

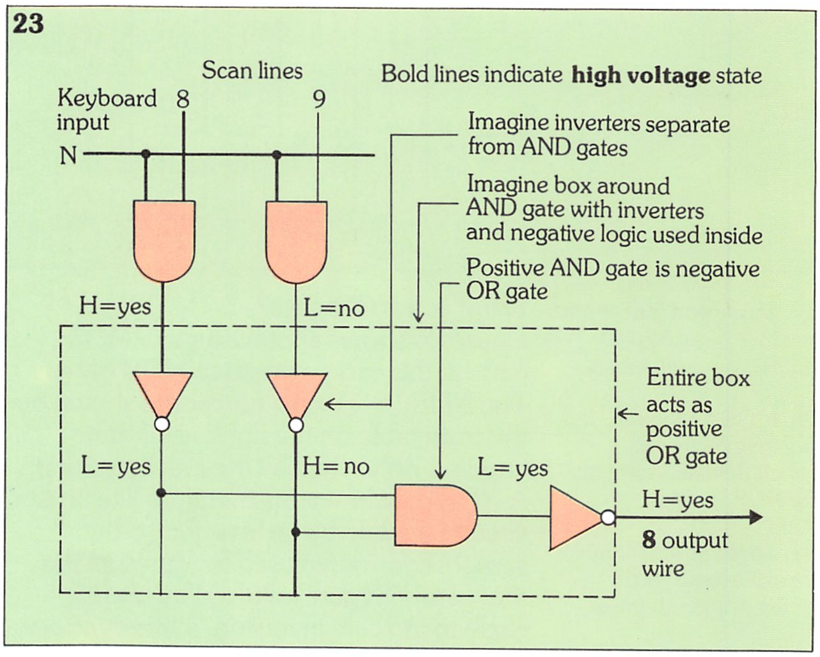
Table 1 Function table

A	B	Q
L	L	L
L	H	L
H	L	L
H	H	H

Table 2 Truth table

A	B	Q
1	1	1
1	0	1
0	1	1
0	0	0


22
Keyboard input lines
Scan lines
1 2 3 4 5 6 7 8 9
N
P
Lines shown bold are **on** (higher voltage) when **5** key is pressed and scan line **5** is on
Number ready signal to controller
1 → 1
2 → 0
4 → 1
8 → 0
0 1 2 3 4 5 6 7 8 9
Internal number lines
Number **5** to display register


23
Scan lines
Keyboard input 8 9
N
Bold lines indicate **high voltage** state
Imagine inverters separate from AND gates
Imagine box around AND gate with inverters and negative logic used inside
Positive AND gate is negative OR gate
H=yes
L=no
L=yes
H= no
L= yes
Entire box acts as positive OR gate
H=yes
8 output wire

Thus we can see that a positive AND gate is a negative OR gate, and a negative AND gate is a positive OR gate. The same relationship applies to NAND and NOR gates.

All the circuitry inside the dotted lines in *figure 23* could be represented by a single OR gate, in which case it would look just like part of the encoder which is shown in *figure 10*. As previously explained however, real circuits are built using NAND and NOR gates because they are easier to construct.

22. How to build the decoder from figure 10 by using positive NAND gates and an inverter, as in a real MOS IC chip.

23. NAND gates feeding a NAND gate can be thought of as AND gates feeding an OR gate.

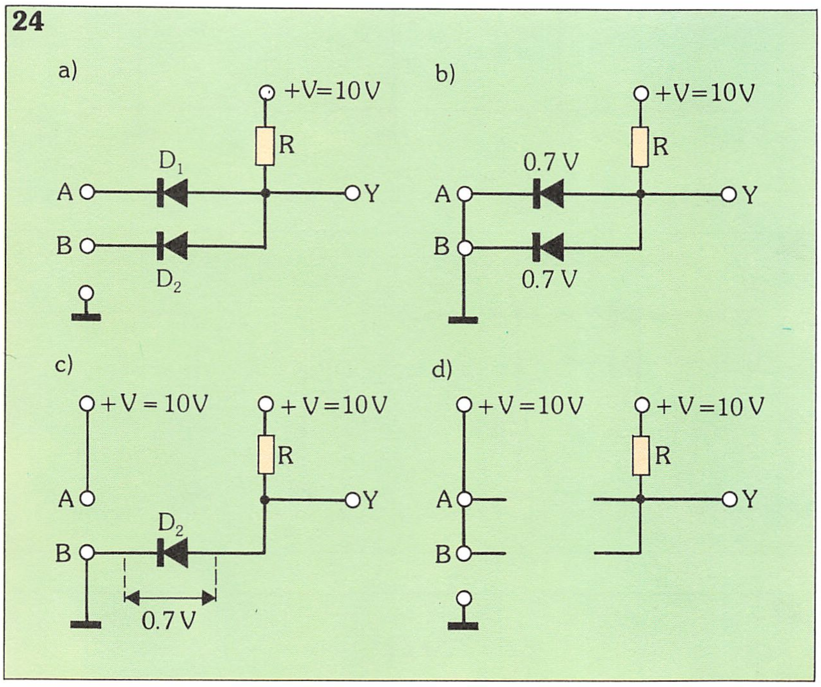
Constructing gates using bipolar transistors

AND gates can be constructed using suitably connected discrete components such as diodes and bipolar transistors (basic NPN or PNP type). These components are explained more fully in the section on *Solid State Electronics*. However its's worth briefly mentioning how a diode works before describing the actual circuits in detail.

A diode is a undirectional conductor. That is, it allows current to flow in one direction only. When the anode is more positive than the cathode (forward bias), the diode has a very low resistance and current flows. When the cathode is more positive than the anode (reverse bias), the diode has a very high resistance and no current can flow through it.

In *figure 24a* the anodes of diodes D_1 and D_2 are connected to + 10 volts via resistor R. If a signal of 0 volts is applied to the cathodes (*figure 24b*) the diodes will become forward biased and current will flow through them. The output at point Y will then drop to about 0.7 volts (the voltage dropped across the very low resistance of the diode). This is considered as logic 0.

24



24. An AND gate made with diodes.

If input A is now raised to 10 volts diode D_1 stops conducting, but diode D_2 continues conducting and point Y stays at 0.7 volts: logic 0. A similar thing happens if point B is put to 10 volts and point A is at 0. *Figure 24c* shows how the reverse biased diode (D_1 in this case) can be considered as an open circuit.

When the cathodes of both the diodes are connected to 10 volts (reverse biased) they both act as open circuits and the voltage at point Y goes up to 10 volts. *Figure 24d* shows how the circuit appears with the cathodes of both diodes connected to 10 volts. This is then logic 1.

To sum up the operation of the circuit, assuming that positive logic has been used (0 = 0.7 V and 1 = 10 V): when one or both inputs are 0, the output is 0; when both inputs are 1 the output is 1. If you draw the truth table for this circuit you will find that it is just the same as the truth table given earlier for a positive logic AND gate.

Making an AND gate using transistors

A transistor can be in three distinct states; it can be off, it can be on, and it can be somewhere in between. More precisely the transistor is off when there is less than 0.7 V between the base and the emitter, and on when the base-emitter voltage is over 0.7 V. In this fully on state the transistor is described as being bottomed. The 'in between' state isn't very important in digital electronics.

Now consider the circuit in *figure 25a* which is composed of three NPN transistors. Applying 0 V (ground) to both A and B inputs, the transistors T_1 and T_2 would both be off and as a result would act as an open circuit (*figure 25b*). The base of T_3 is at a positive potential which causes it to operate and therefore output Y is grounded (0 V), i.e. it is at logic 0.

If 10 V is applied to input A while the 0 V signal remains applied to B, transistor T_1 is bottomed, while T_2 operates as before – so output Y is again at 0 V. Therefore we can see that for the output to be 1, i.e. 10 V, both T_1 and T_2 must be bottomed.

In this case, in fact, the base of T_3 would be at 0 V (earth), T_3 would behave as an open circuit and the Y output would rise to take the value of 10 V (+V).

Through the use of transistors, we have managed to make another positive logic AND gate.

Using transistors and diodes to make OR gates

An OR gate can also be made using diodes and transistors. If a voltage of 10 V is applied to point A (*figure 26a*) and 0 V to point B, diode D_1 will conduct and diode D_2 will be reverse biased. As a result point Y will be 10−0.7 volts (0.7 V is the voltage dropped across the diode). This then is the logic 1 state. The same situation applies if point A is 0 V and point B is 10 V or if both are at 10 V. When both inputs are at earth potential neither diode conducts and point Y will be at earth potential (0 V) which is the logic 0 state. To summarize the way a diode logic OR gate works, when one or both of the inputs are high (positive) the output is high. When both inputs are low (earth) the output is low.

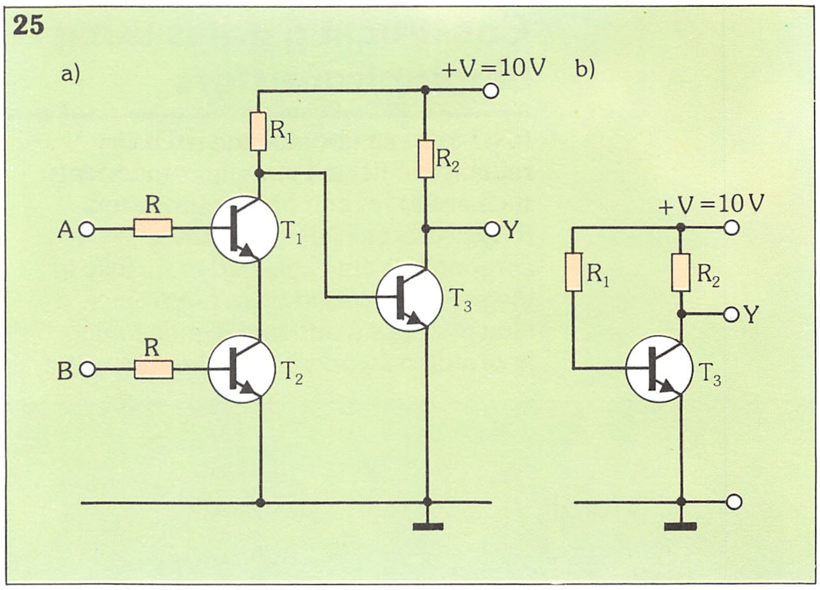
Transistors connected as shown in *figure 27* will also act as logic OR gates. If point A is connected to a positive voltage and point B is connected to ground then T_1 will conduct and T_2 will be off. In this condition point Y will be close to 10 V. Point Y would also be 10 V if point A was at ground potential and point B was at a positive potential, and indeed if both A and B were positive. Finally when both inputs are grounded then point Y will be at ground potential (0 V). *Figure 28* shows the equivalent switch circuits for the transistor logic OR gate under the various input conditions.

NAND and NOR gates using transistors

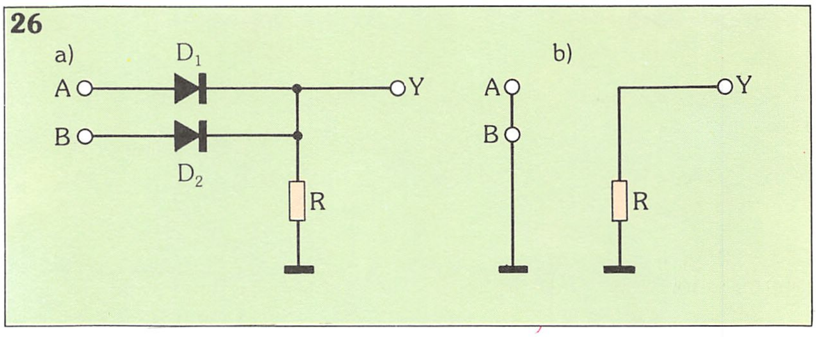
As with AND gates and OR gates, NAND and NOR gates can be made using diodes and transistors.

Figure 29a shows a NAND gate. If 10 volts (logic 1) is applied to point A and 0 volts (logic 0) to points B and C then transistor T_1 will be trying to conduct but transistors T_2 and T_3 are held off and no current flows to ground. Point Y is then at logic 1. The same situation applies if *any* one or more of the inputs is at O V. However, when all the inputs are at 10 V then all the transistors conduct and point Y is held at logic 0.

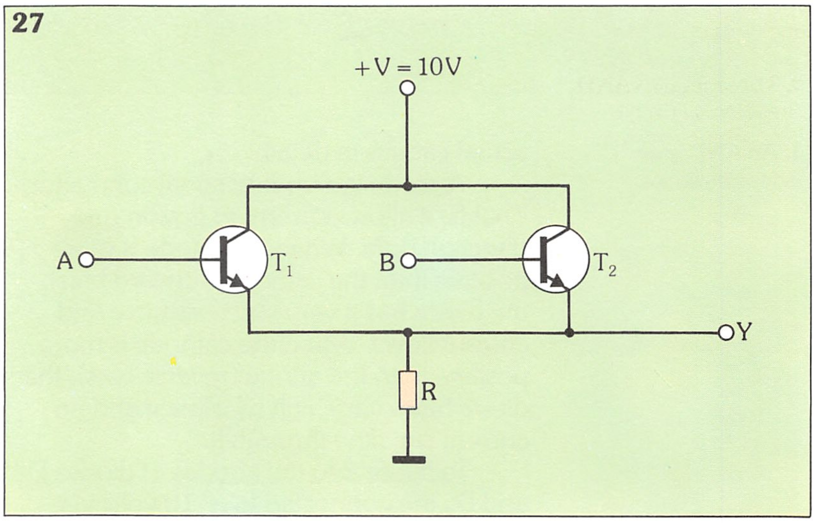
25



26



27

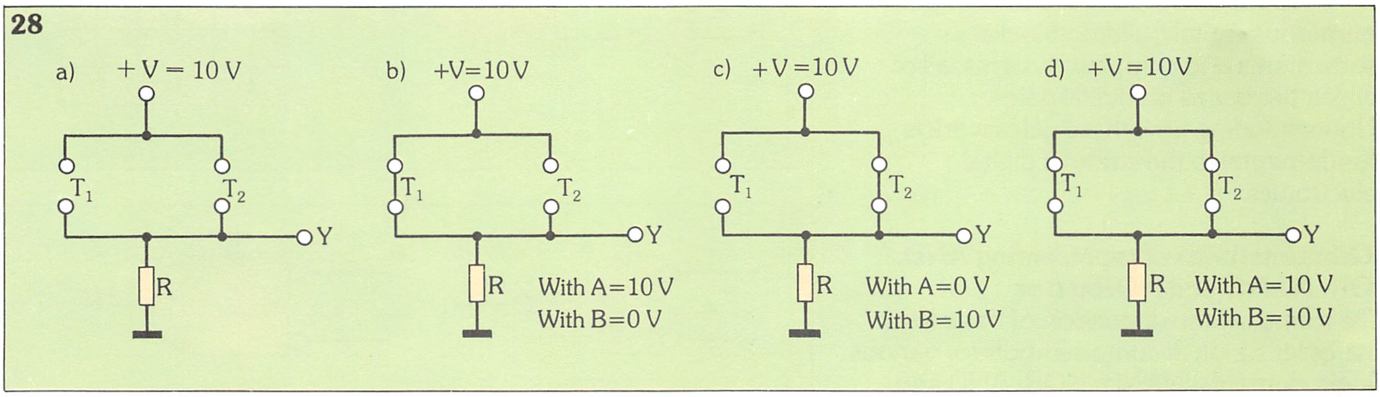


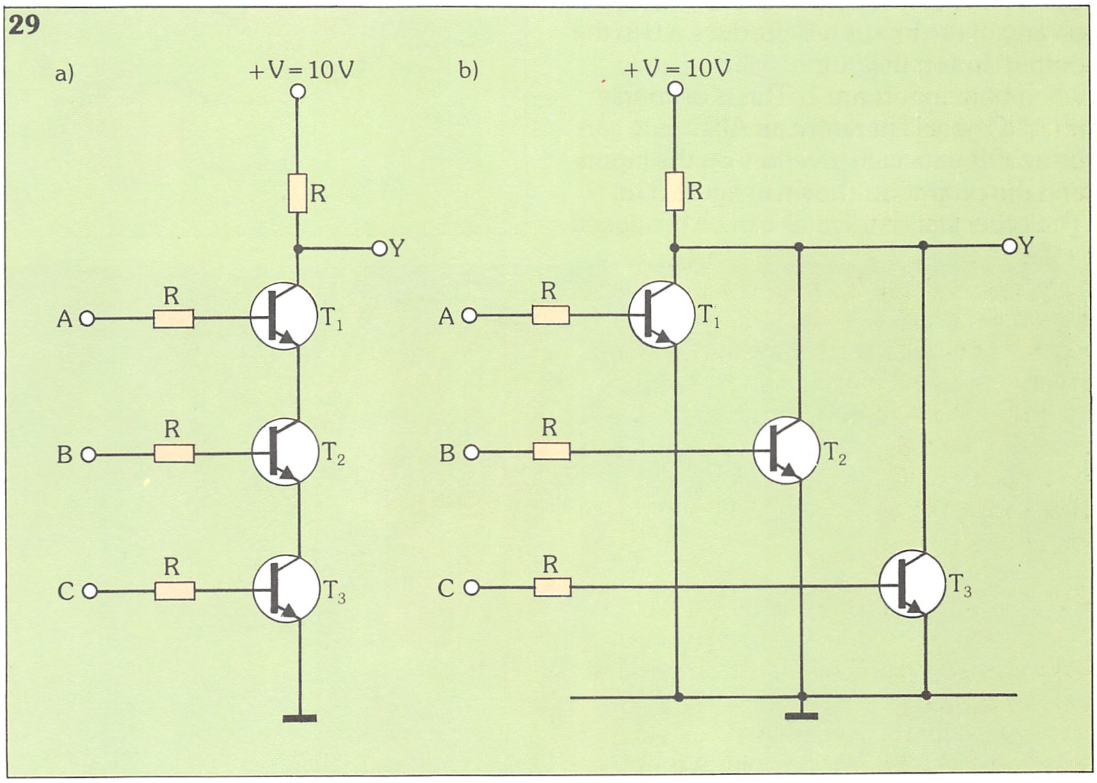
To summarize the action of this circuit it can be said that a low (0) on any of the inputs causes the output to be high (1), whereas if *all* the inputs are high (1) the output will be low (0). This then is the NAND function made from transistors.

The transistor logic NOR circuit is

shown in *figure 29b*. If a voltage of 10 volts (1) is applied to point A, with both the other inputs B and C at 0 volts (0), then transistor T_1 will conduct causing point Y to go to a very low voltage (0). Point Y will remain low (0) as long as one or more of the inputs is at 10 volts (1). If all the inputs are at 0 volts (0), none of the transistors conduct and point Y will be at 10 V (1).

Summarizing the action of this circuit, it can be seen that when one or more of the inputs is high (1) the output will be low (0), while with *all* inputs low (0) the output will be high (1). This of course is the NOR function.





25. An AND gate made using transistors.

26. An OR gate made using diodes.

27. An OR gate made using transistors.

28. Equivalent circuits for figure 27 for various conditions of T_1 and T_2.

29. Three-input NAND (a) and NOR (b) gates made using transistors.

It should now be clear that digital systems can make decisions. The designer of a system just breaks down complex decisions into smaller ones that are simple enough to be made using gates. There are five basic kinds of gate to choose from: AND, OR, NAND, NOR and NOT.

In reality the range of gates is often reduced depending on the particular type of integrated circuit chosen for the system design.

Often, as in the example of the encoder, the choice is limited to just NAND, NOR and NOT gates. However, as you have seen, by using suitable combinations these gates can be made to act like AND gates and OR gates.

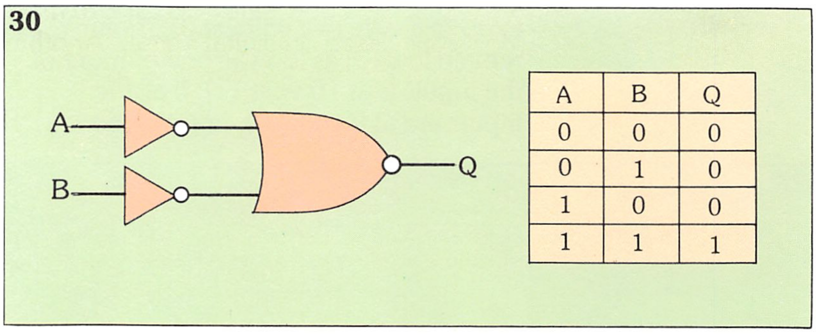
In later chapters you will become more familiar with gates. We will show how system requirements can be analysed in order to select the most efficient combination of available gates, and explain the differences between families of integrated circuits.

We will also look at the workings of numerous digital building-blocks, subsystems and complete systems, all of which are based on digital gates. Understanding how these gates work is fundamental to the study of digital electronics.

Other methods of representing AND, OR, NAND and NOR gates

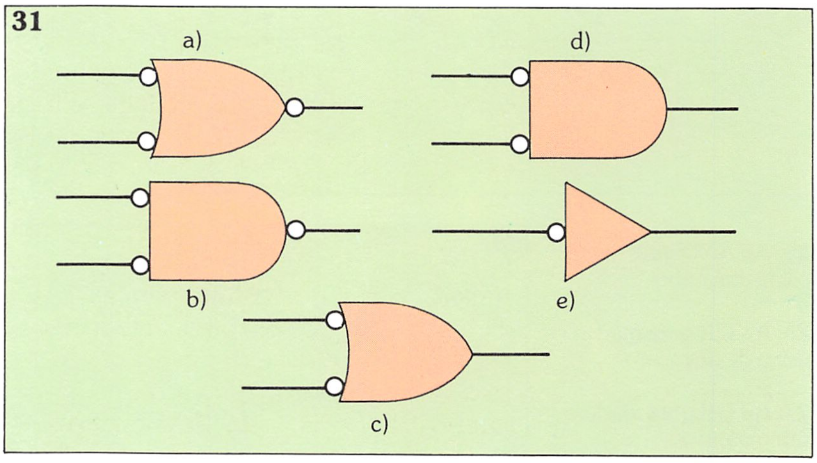
To round off the discussion of basic logic gates let's look at some symbols for various logic elements, NAND, NOR, AND and OR. The circuit in *figure 30* shows that a 0 on any of the inputs will produce a 0 in the output. In fact the output will be 1 only when both inputs are 1. This is of course an AND gate. Therefore an AND gate can be an OR gate with inverters on the inputs and the output, as shown in *figure 31a*. The other standard gates can be produced as shown. *31b* is an OR gate, *31c* a NAND gate, *31d* a NOR gate and *31e* is an inverter (NOT gate).

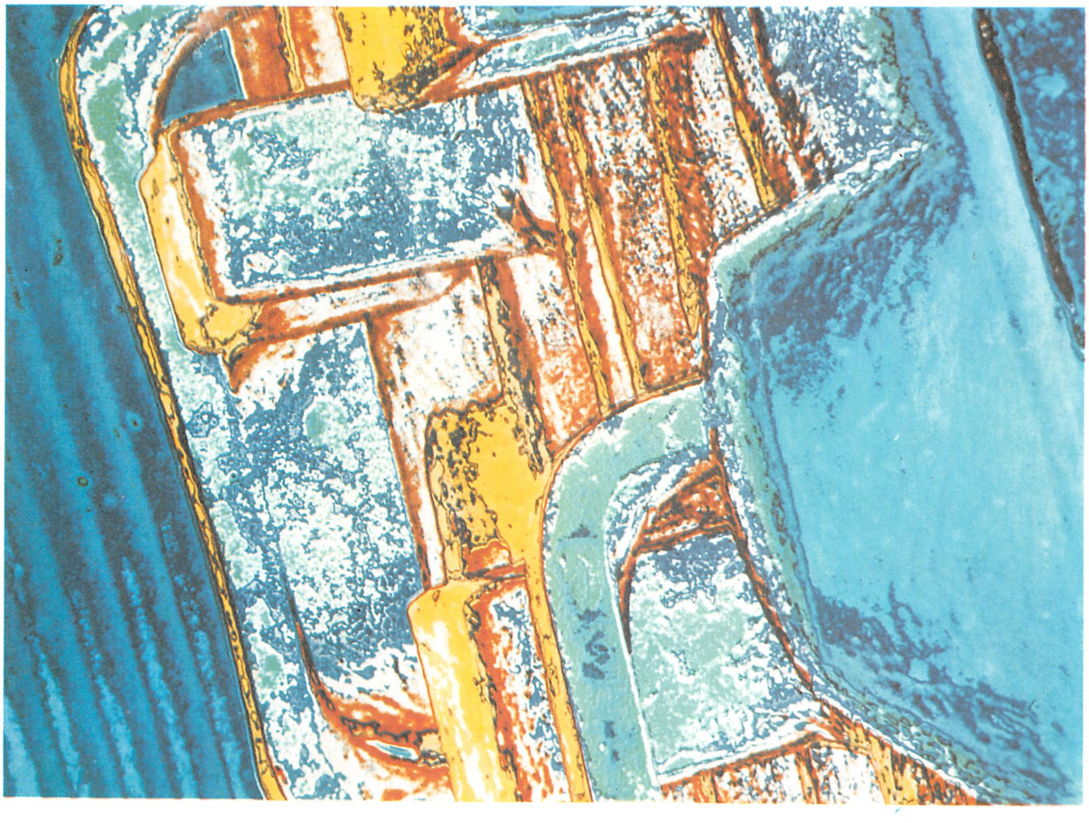
30



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

31





30. A digital circuit that acts like an AND gate.

31. Alternative methods of making and representing AND(a), OR (b), NAND (c), NOR (d), and NOT (e) gates.

Part of a monolithic digital IC magnified by a factor of 110 using an electron microscope. An IC can have more than 50,000 logic gates. (Photo: IBM).

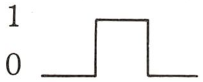
Gating a digital signal

We have seen how gates are used to make decisions based on digital signals. Another equally important job they can do is to control or 'gate' a digital signal.

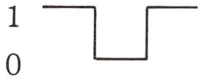
But what is a digital signal? It is a series of pulses with just two levels. The classical digital signal (*figure 32*), is a binary signal which is defined as a voltage or current that carries information in the form of two distinct states which are separated from each other by discrete time intervals. One state is logic 0, which normally corresponds to 0 volts or earth potential, the other state is logic 1 which in general corresponds to the supply voltage. When a current signal is used logic 0 corresponds to zero amps while logic 1 is usually 20 mA (positive logic).

Operations on a single digital signal

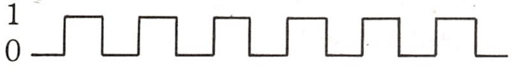
A digital circuit is one that operates on a digital input signal, changing it to a digital output signal as shown in *figure 33*. Let's go a little further into this explanation. Suppose the input is a series of pulses. A clock pulse, for example, is a series of regular pulses changing from 0 to 1 and back again. This is generally used to control and regulate the operation of digital circuitry. The time taken to complete the transition is the clock period. A clock pulse of this type is shown below:



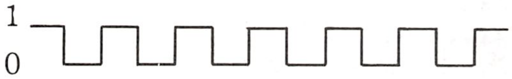
Alternatively a clock pulse can be defined as starting from 1, going to 0 and back again, as shown below:



A sequence of clock pulses can be represented as



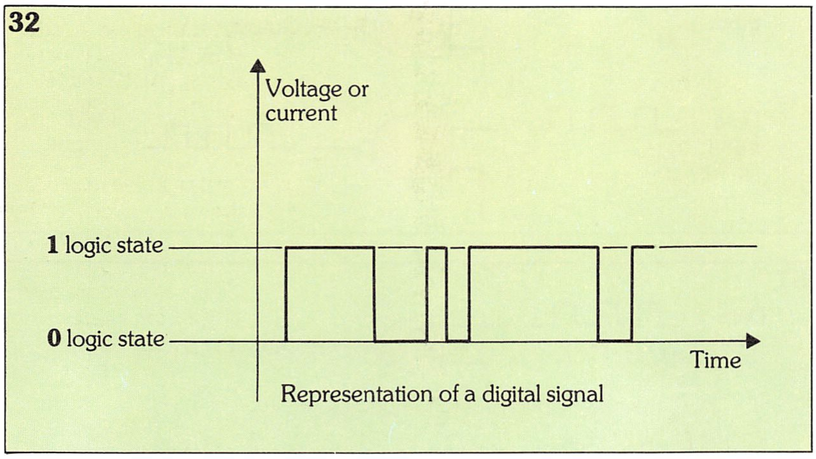
or as

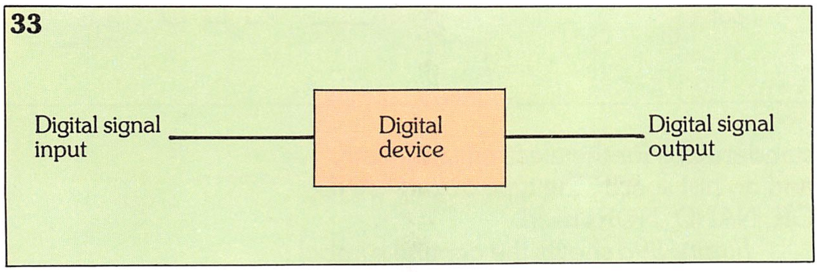


Note that the sequences are inverse, like the two individual pulses above. What then happens to this sequence of clock pulses when it passes through different digital devices?

32. Example of a digital signal. It has 2 levels (0 and 1).

33. Generalized block diagram of a digital device.





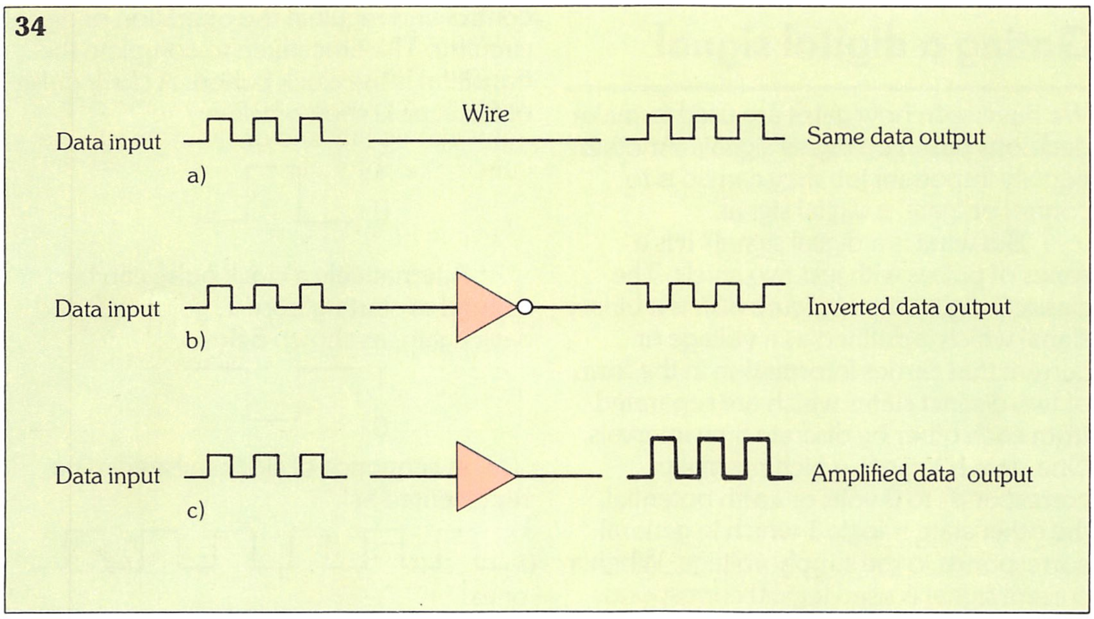
Transmitting and gating

The simplest operation one can carry out on a sequence of pulses is that of transmitting it unchanged. This is done by a wire (*figure 34a*). The next simplest is that of inversion; using an inverter (*figure 34b*). A third simple operation is that of amplifying the pulses by means of a buffer, a driver or an amplifier (*figure 34c*).

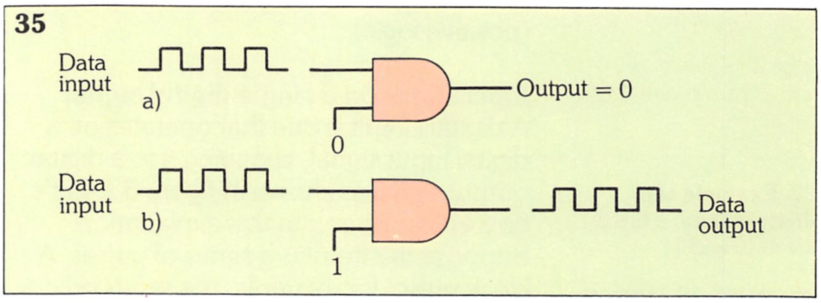
Each of the three operations just mentioned involves handling an input signal in a consistent way; there is no variable control placed on the signal. **Gating** a signal means controlling it, turning it on or off, depending on a particular set of circumstances. Four simple gates are used: AND, NAND, OR and NOR.

A sequence of pulses through an AND gate

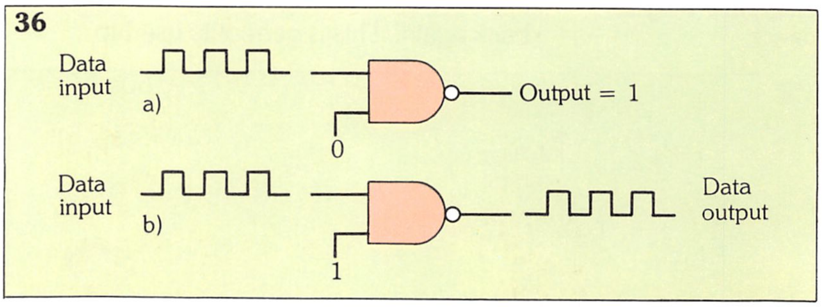
When a gating signal to an AND gate is at logic 0 the input data, such as a sequence of clock signals, cannot pass through the gate. This is shown in *figure 35a*. When, on the other hand, the gating signal to the AND is at 1, the input data is transmitted directly to the output (*figure 35b*).



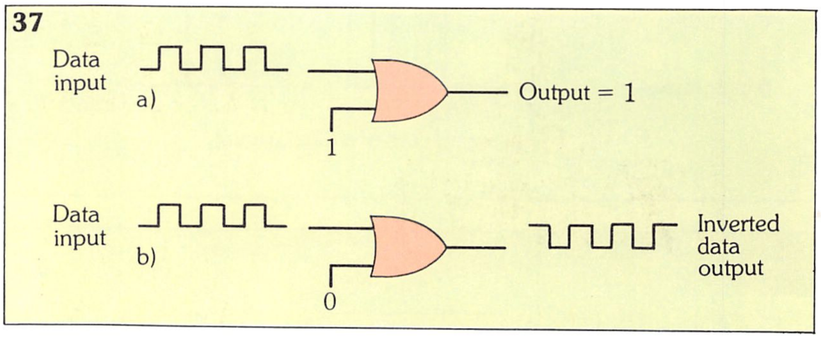
34. A series of pulses transmitted via a wire, inverter, and amplifier.



35. A gated signal using an AND gate: signals can be blocked (a) or gated (b). The output signal is the same as the input signal.



36. Gating a signal using a NAND gate: the signals can be blocked (a) or gated (b). The output signal is inverted.



37. Gating a signal using an OR gate: signals can be blocked (a) or gated (b). The output signal is the same as the input signal.

A sequence of pulses through a NAND gate

A NAND gate (Negated AND gate) operates in a similar way to the AND gate. When the gating signal is at logic 0, the input data is blocked (*figure 36a*). When the gating signal is at logic 1, the output data will be the input data inverted. This is represented with a series of clock pulses in *figure 36b*.

A sequence of pulses through an OR gate

In an OR gate, the input data is blocked by the application of logic state 1 at the gating signal input (*figure 37a*) which gives a logic state 1 at the output. Logic 0 causes the input data to be transmitted to the output (*figure 37b*).

A sequence of pulses through a NOR gate

A NOR (Negated OR) gate bears the same relation to an OR as a NAND to an AND. As for an OR, the application of a gating signal 1 prevents the transmission of the data to the output (*figure 38a*). A logic 0 gating signal will output the inverted input data as shown in *figure 38b*.

Three-input gates

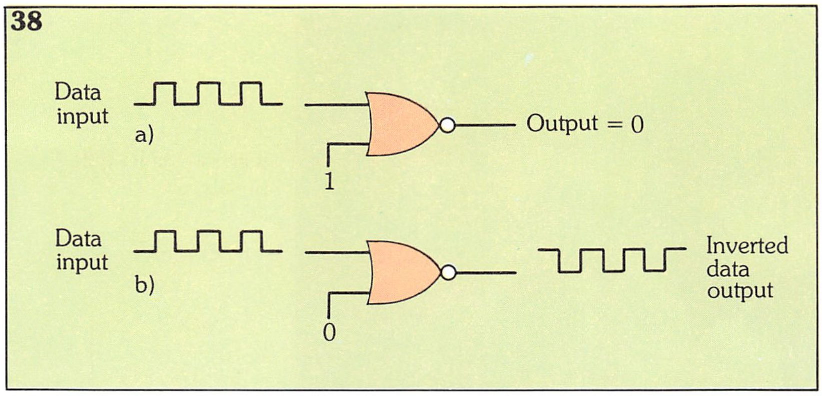
Logic devices with more than two inputs can be used for gating a digital signal. *Figure 39* shows examples of the conventional representation of gates. These are standardized for the electronics industry and do not specify the type of gate (AND, OR, NAND, NOR) used.

Figure 39a shows the general symbol

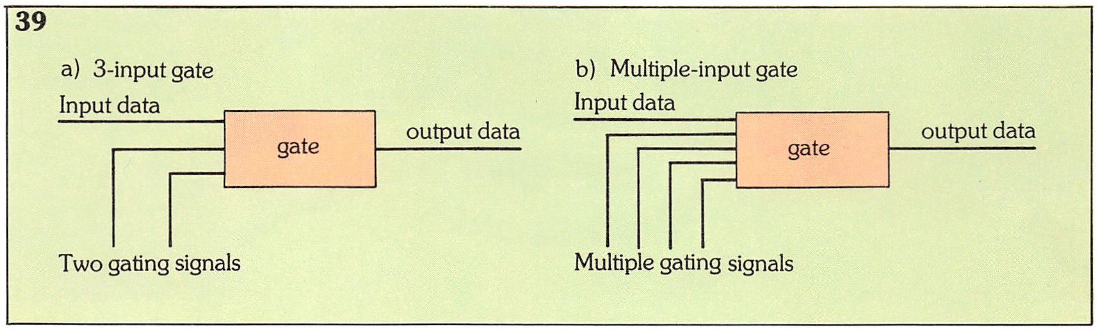
for a gate with 3 inputs. There is one single data input and two gating inputs. According to the nature of the gate and the logic states of the two gating signals, the data input can be blocked or sent through the gate to the output line.

Multiple input gates

Figure 39b represents a multiple input gate. These are used very extensively in most digital systems. Generally only AND and NAND gates are available with multiple inputs.

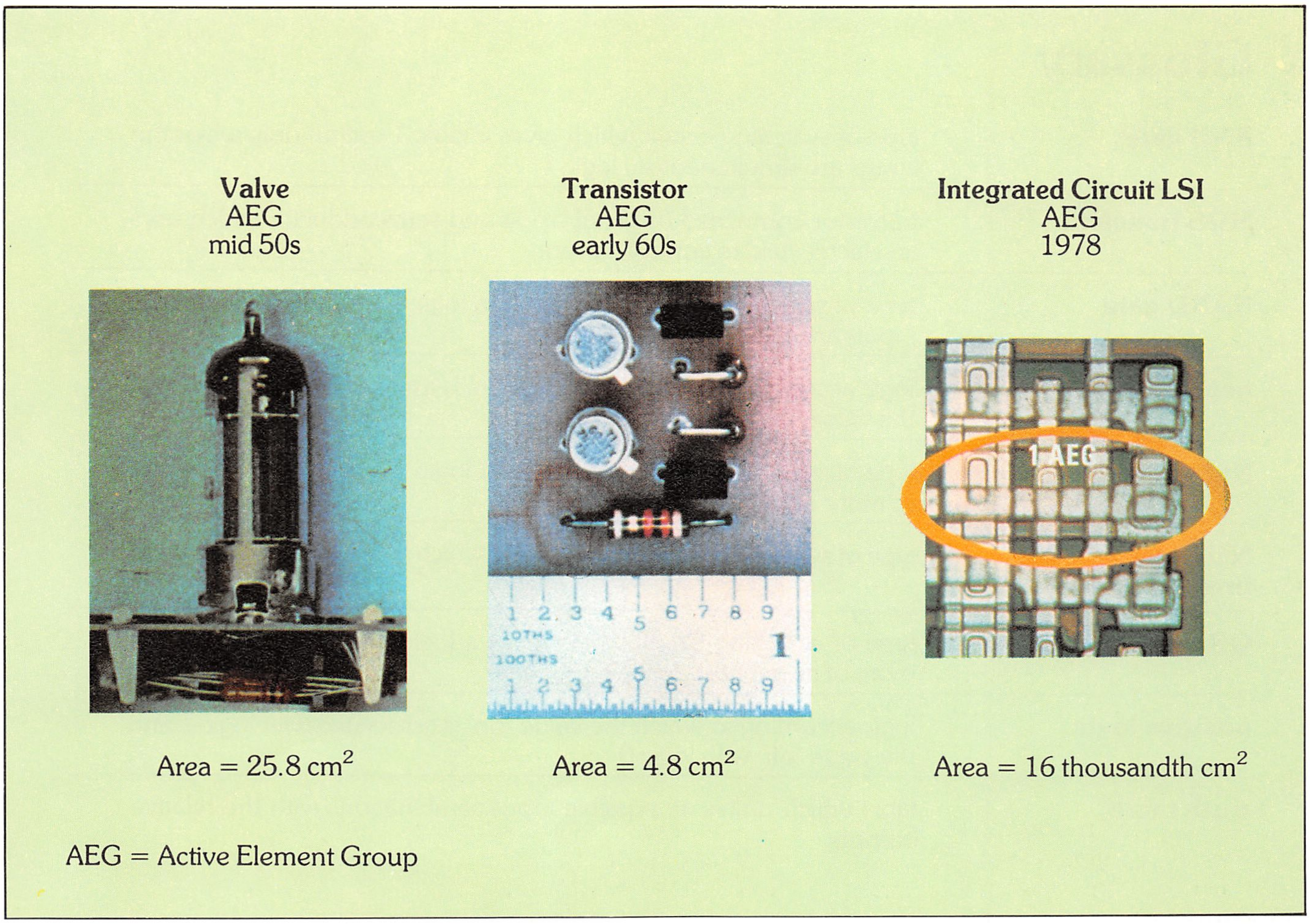


38. The gating signal applied to a NOR gate. The pulse shown can be blocked or inverted.



39. Normal representation of gates with (a) three inputs (b) multiple inputs.

The advance of semiconductor technology.



These technicians are using computer-aided design to configure standard logic cells into special circuits on a single IC.
(Photo: courtesy S.G.S.)

Glossary

AND gate	type of switching circuit which gives a logic 1 output only when the inputs are simultaneously logic 1
MOS transistor	transistor composed of metal oxide and semiconductor which uses an electric field to control current
NAND gate	type of switching circuit which gives a logic 1 output only when the inputs are simultaneously logic 0
negative logic	logic according to which the value 'on' is associated with logic 0 and the value 'off' with logic 1
NOR gate	type of switching circuit which gives a logic 1 output only when one or more of its inputs is logic 0
NOT gate or inverter	type of switching circuit the output of which is the inverse of its input
OR gate	type of switching circuit which gives a logic 1 output when one or more of its imputs is logic 1
positive logic	logic according to which the value 'on' is associated with logic 1 and the value 'off' with logic 0
truth table	table which unites all possible input combinations with the relative outputs

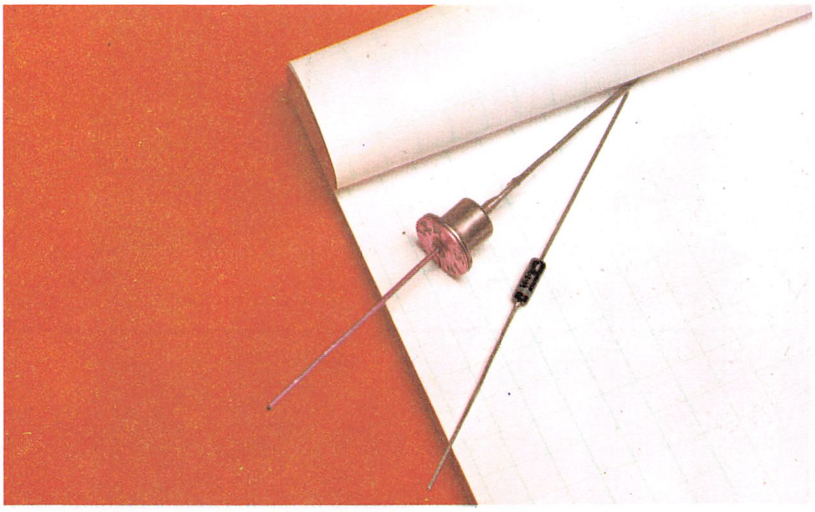
4

SOLID STATE ELECTRONICS

Introducing diodes

Diodes: the simplest type of semiconductor

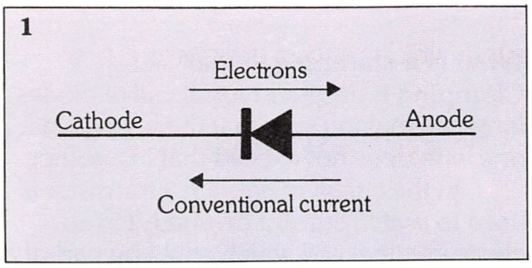
Having discussed in general terms what semiconductor devices can do let's now look at some specific examples. **Diodes** are the logical starting point for two reasons. They are the simplest type of semiconductor device and the basic understanding of how they work can be applied to other types such as transistors, integrated circuits and even large-scale integated circuits (LSI).



Two different diodes. The larger one is used as a rectifier.

What is a diode?

Diode means 'having two electrodes' and the diode is simply a package with two terminals or wires. The diode's most important function is to act as a one-way valve for the passage of electrons. It allows electrons to pass in one direction only, from the **cathode** (negative terminal or emitter) to the **anode** (positive terminal or collector), but blocks their passage in the other direction. *Figure 1* shows the direction of electron flow in a symbol for a diode. Remember that conventional current flow is in the opposite direction. The diode is therefore basically a switching, rather than varying, device.



1. The schematic symbol for a diode. The electron flow is in the opposite direction to the large arrowhead that forms part of the symbol.

Most semiconductor circuits operate on direct current and one of the main applications of diodes is to convert alternating current (ac) into direct current (dc).

Figure 2 illustrates the normal ac wave form 'A'. This type of current will not allow the dc motor shown here to work. If a diode is inserted into the circuit as shown, it will allow the top half of the wave form to pass, but not the bottom half. It is as if everything below the broken line (0 V) were rubbed out. All the current pulses which pass are sent in the same direction.

This function is called **rectification.** In this case it is half wave rectification, as we are left with only half the original wave form. Diodes built specifically for this purpose are called rectifiers. They are capable of high power dissipation at low frequencies.

Obviously the current pulses which remain after rectification are not a very regular source of power. So we can include a capacitor to smooth out the waveform to that shown by B in *figure 2*.

A capacitor is a bit like an electron storage tank which releases electrons in a steady stream. To understand how a capacitor does this we'll return to our familiar water example. Look at the water tank illustrated in *figure 3*. Imagine that the man keeps on pouring in buckets of water. The water is entering in spurts or pulses while at the same time it flows out of the other side of the tank in a fairly constant, smooth stream. Capacitors store and release energy in much the same way.

A low power application

As an example of low power application we will look at a crystal diode. The earliest crystal diodes were used in radios of the twenties and thirties, where natural lead sulphide (galena) crystals functioned as semiconductor diodes long before the word 'semiconductor' was thought of. More advanced versions are in use today.

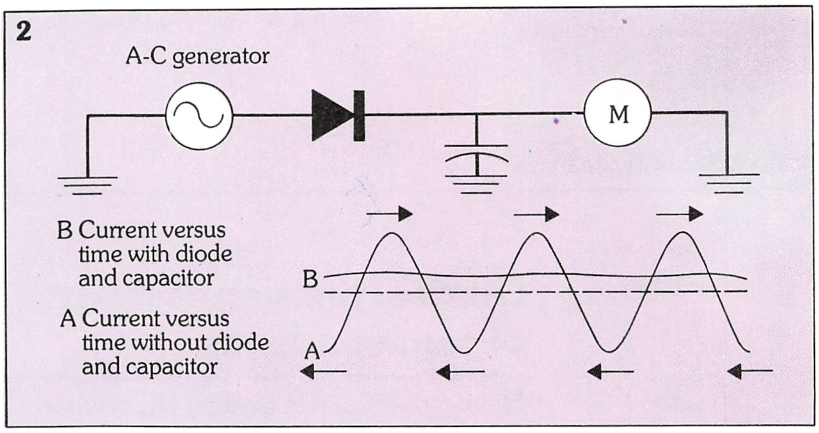
Figure 4 illustrates an AM radio transmitting and receiving system. Underneath each stage of the system you will notice the appearance of the waves at that point.

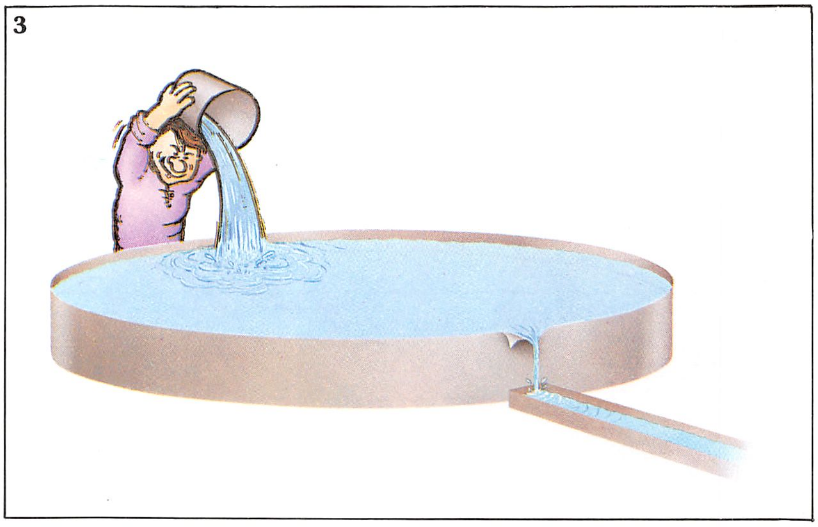
In *Solid State Electronics 2* it was shown how a low frequency microphone signal is used to modulate the amplitude of high frequency waves produced by an oscillator. The modulated signal is first converted into radio waves suitable for a broadcast aerial and then reconverted into electrical waves by the receiving aerial. In our example a crystal diode with moderate power and frequency ratings is used to cut off one side of these modulated waves, in the same way as the rectifier did for the motor in the preceding example. This rectified signal drives the headphones.

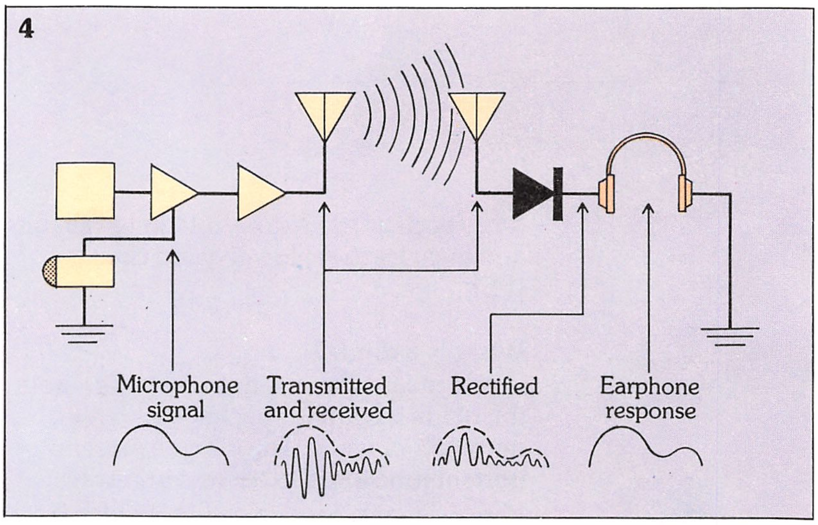
The headphones in turn 'average out' the radio frequency pulses in much the same way as a capacitor does; the ear phones are simply unable to respond to every little pulse of the high frequency carrier waves. The resulting sound they produce is a fairly accurate replica of the microphone signal.

The microphone is said to *modulate* the signal because it modifies the amplitude of the high frequency waves emitted by the oscillator. The reverse process which the diode carries out here is therefore called demodulation or *detecting*. Diodes employed for this purpose are called detectors.

Modern diodes used as detectors have the same function as the old galena crystal but they are smaller, cheaper, far more reliable and they can take more power. Since receiving ordinary AM broadcasts and driving earphones only requires moderate power and moderate frequency ratings, these diodes are called 'general purpose' diodes.


2
A-C generator
M
B Current versus time with diode and capacitor
A Current versus time without diode and capacitor
B
A





What is a clamping diode?

Clamping is another typical use of diodes. It means making sure that the voltage in one wire does not exceed that in another.

In the circuit in *figure 5* a transistor is used to switch current on and off in an electromagnet coil which could be part of a

3. The action of this water tank illustrates the smoothing effect of a capacitor.

4. AM radio transmitting and receiving system.

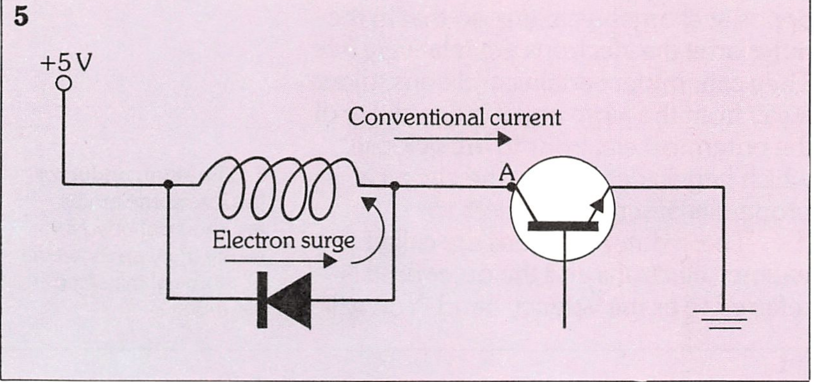
2. A primary application of diodes is converting alternating current into direct current.

motor or a relay. Like all coils it has **inductance.** Once the electrons in the coil start moving they do not want to stop even after the transistor is turned off. When it is turned off by blocking the current at point A the electrons keep coming and pile up, causing a high voltage. If this becomes too high the current could break through the transistor at a dangerously high voltage and damage it.

The solution is simple. A diode is inserted across the coil to act as a safety valve. Because of its one way action the diode will not allow the electrons coming from the power supply to make a detour round the coil and enter the transistor. It will, however, allow electrons to bypass the coil in the opposite direction. So, if the voltage generated by the electrons at A is higher than that of the power supply (5 V), the diode will allow the electrons to pass and drain the excess build-up away from the transistor. The collector of the transistor is in this way 'clamped' at 5 volts. The voltage can be below 5 volts, but it will never be much above it.

To sum up: diodes can be used for rectifying, detecting and clamping as we have shown. Another important use is in the building of logic gates, dealt with in depth in the *Digital Electronics* section.

5



5. A diode can be used as a kind of safety valve to protect a transistor from damage due to surges of high voltage.

Glossary

capacitor	electrical device which is commonly used to smooth out irregular pulses in electrical current, allowing a more constant flow of electrons
clamping	the use of a diode to prevent the voltage in one wire from exceeding the voltage in a second wire
detector or demodulator	carries out the reverse process of a modulator (see below)
inductance	characteristic of an electrical circuit in which the electro-magnetic field generated by the flow of current generates a resistance to the alteration of the current, by producing a back EMF proportional to the rate at which the current changes
modulator	an amplifying-type circuit the output of which is a copy of oscillating electrical waves at its input, except that the amplitude (height) of the output waves is modulated (controlled) by a second input
diode	semiconductor device which permits electrons to flow through it in one direction only
rectification	the most straightforward use of a diode: the conversion of alternating current to pulses of direct current
rectifier	a diode built specifically for the purpose of rectification

SOLID STATE ELECTRONICS

Semiconductor materials

Properties of semiconductor materials

The working of transistors, diodes, thyristors and all other semiconductor devices depends on the controlled flow of current through semiconductor material. It was mentioned earlier that these semiconductor materials can be of two types, p-type and n-type, which are combined to make the actual devices. But in order to understand how these devices work you need first to look at their basic atomic structure.

A simplified model of an atom (*figure 1*) shows a central, positively charged nucleus surrounded by negatively charged electrons which rotate round the nucleus in fixed orbits. The nucleus itself is made up of two types of particle: protons which are positively charged and neutrons, which are neutral.

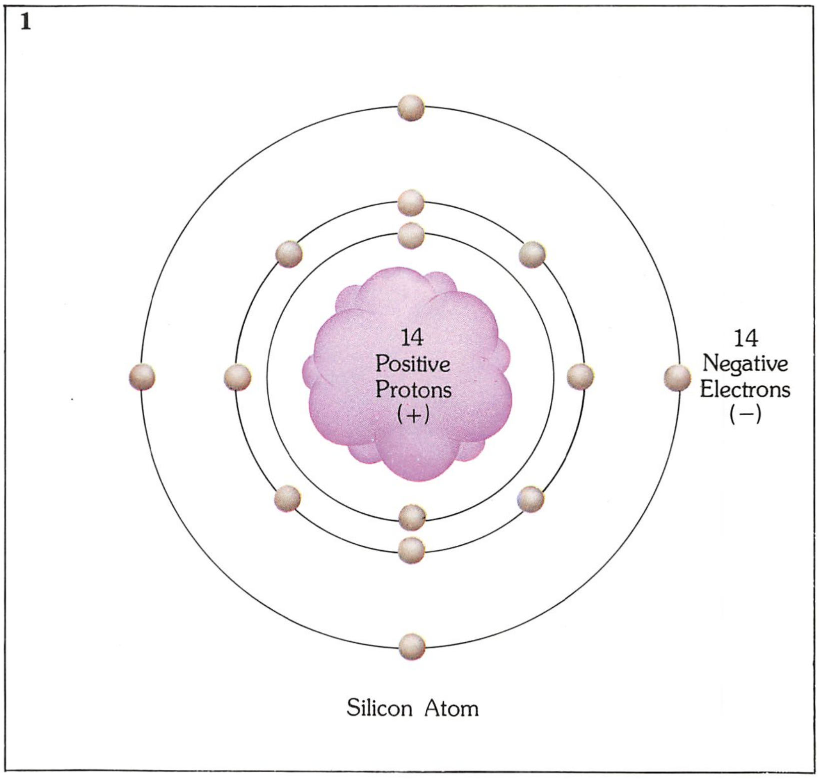
The number of protons inside the nucleus is equal to the number of electrons circling round it and this number is fixed for each particular element. For instance, silicon has 14 protons and 14 electrons while germanium has 32 protons and 32 electrons. *Figure 1* is a diagram of a silicon atom.

The negative charge of an electron is equal and opposite to the positive charge of a proton. It therefore stands to reason that when the number of electrons is the same as the number of protons they cancel each other, and the charge on the atom is neutral.

A very basic rule of nature is that *like, charged particles repel, unlike particles attract.* So it follows that two protons will repel each other as will two electrons, while a proton and a neutron attract each other.

Now let's look at the electrons orbiting round the nucleus. They are bound to the nucleus by their opposite charge. Those in the inner orbit, closest to the nucleus, are very tightly bound but as they get further away the attraction between the opposite charges weakens, so that in the outer orbit the electrons are relatively free. They can, under certain conditions, move away from the atom and it is this ability of the outermost electrons to break loose which largely determines the electrical properties of semiconductors.

These outer electrons are called **valency** electrons and the outer orbit is referred to as the valency band. You will see from *figures 2* and *3* that silicon has four valency electrons and three orbital levels. Although germanium also has four valency electrons it has more electrons overall, giving it four orbital levels.



1. The semiconductor lattice. Some atoms are shown shaded for clarity, they are however all atoms of the same element.

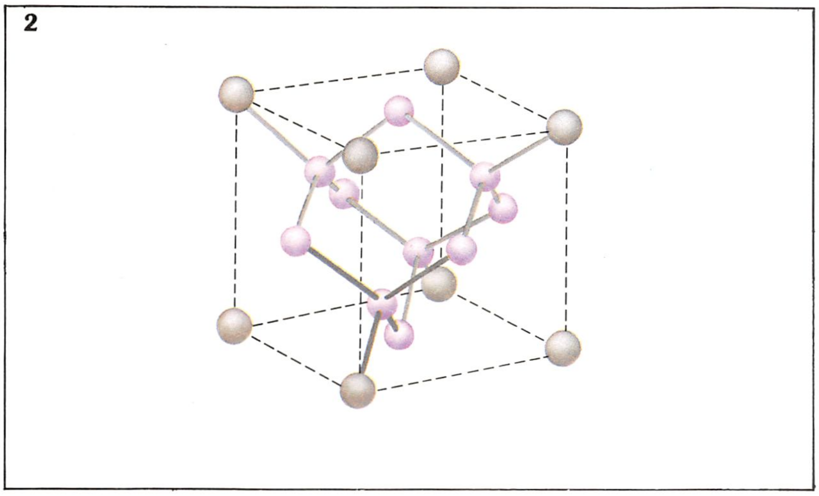
Crystal structure

When the atoms of an element are brought close to one another, as in the case of a solid, they often bind together in a uniform, three-dimensional structure to form **crystals.** The shape of these crystals varies according to the elements involved, and is determined by the element's atomic structure and, above all, by the valency electrons.

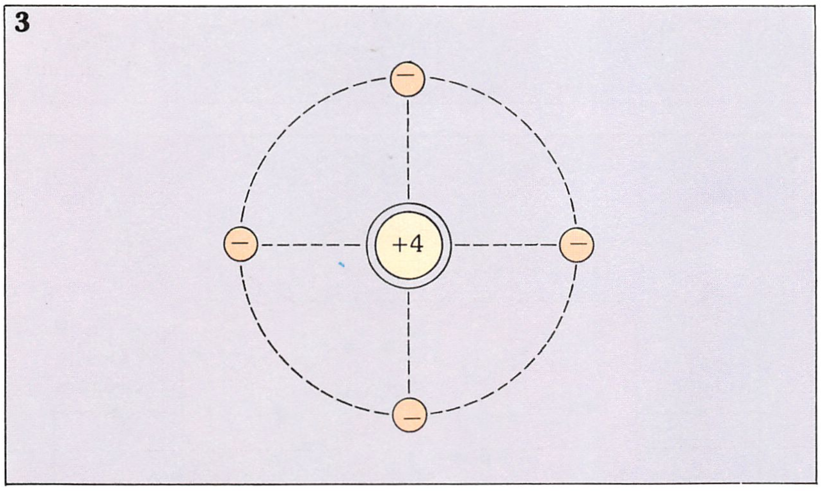
Knowing that like particles repel, you will probably be wondering why the atoms bind together. This brings us to another important aspect of atomic behaviour. Atoms have a tendency to organize themselves into groups and share electrons with neighbouring atoms to gain a total of eight electrons in their outer shell. Having eight electrons in the outer orbit of an atom will render it stable, a state which atoms 'desire'. This 'desire' binds atoms together to form crystal structures. For example, silicon and germanium both have four valency electrons and require four more to make the number up to eight. To get eight electrons in the outer shell, silicon shares its valency electrons with four neighbouring silicon atoms and at the same time shares a valency electron from each of the four neighbouring atoms. Bonds formed by the sharing of electrons are called **covalent bonds.**

2. Diagram of a silicon atom.

2



3



3. Simplified diagram of a silicon or germanium atom, showing the four valency electrons.

Figure 2 shows the result, which is called a **crystal lattice.** Germanium and silicon have similar crystal structures with all the atoms equidistant from each other.

Before moving on to how silicon acts as a conductor there is one more property of the atom which we should remember. Going back to *figure 1* we see that ten of the electrons are in the two shells closest to the nucleus. These are tightly bound to the nucleus, unlike the valency electrons. With the nucleus and the two inner shells we have ten electrons and fourteen protons, giving a net charge of +4. In the outer shell we have four electrons, giving a charge of −4. So the −4 of the outer shell balances the +4 of the core leaving the whole atom electrically neutral (see *figure 3*). However, should these electrons become disassociated from the nucleus we would have a nucleus with a +4 charge and four negatively charged electrons.

Pure or intrinsic semiconductors

A very pure semiconductor material is called an **intrinsic** semiconductor (in practice, however, absolutely pure crystals do not exist and nearly pure materials are called intrinsic). At absolute zero (0°K or −273°C) pure germanium and silicon are exactly as shown in *figure 4*. All the valency electrons are firmly held in their own atoms, and their covalent bonds are stable. The electrons are not free to move in the crystal structure, and so they can't conduct electricity; an intrinsic semiconductor at absolute zero is an insulator.

So how do silicon and germanium become good conductors? The answer is through heat and the addition of impurities. Let's first see how heat can turn an insulator into a good conductor.

The effects of heat

As the temperature increases valency electrons are given more energy and move faster. When an electron reaches a certain level of energy it is able to break free from the bond holding it to the original atom. This electron is then free to move through the crystal structure and becomes a **free electron.** If a voltage is applied to the semiconductor these free electrons then form the electric current.

When an electron moves out of its orbit it leaves an incomplete covalent bond or a **hole** as shown in *figure* 5. These holes also act as current carriers. Imagine that an electron escapes and leaves a hole at point A, say. This hole can then be filled by another electron which in turn leaves a hole at point B. So the electron has moved from B to A and the hole has moved from A to B.

To demonstrate this, think what happens when you go to the cinema and find that all the seats have been taken, apart from one in the middle of the row (*figure* 6). You can choose to go directly to the seat (thereby acting like a free electron and occupying a hole). Alternatively, you can ask everyone to move up a seat (which is equivalent to the hole moving towards the electron).

Whenever there is a hole there is an associated positive charge, since the net positive charge (+4) of the nucleus is now greater than the total negative charge of the three remaining electrons. So a hole is a **positive charge carrier.**

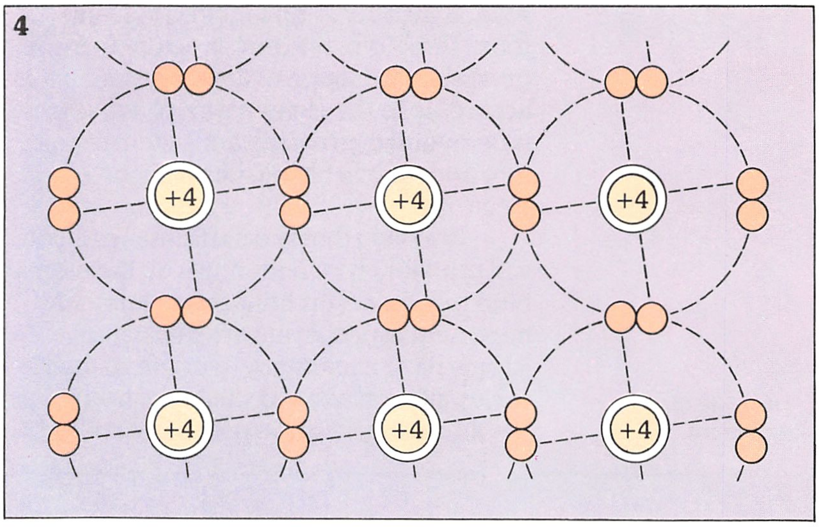
At normal ambient temperatures (about 25°C), thermal energy is sufficient to generate a large number of free electrons and holes in an intrinsic semiconductor. Because of this, materials such as germanium and silicon are reasonable conductors of electricity. They are not such efficient conductors as copper, but they are certainly far better than insulators such as rubber, for example.

In short, an important characteristic of intrinsic semiconductors is that their resistance decreases as temperature increases, because the number of charge carriers, that is, free electrons and holes, goes up, and so the resistance goes down.

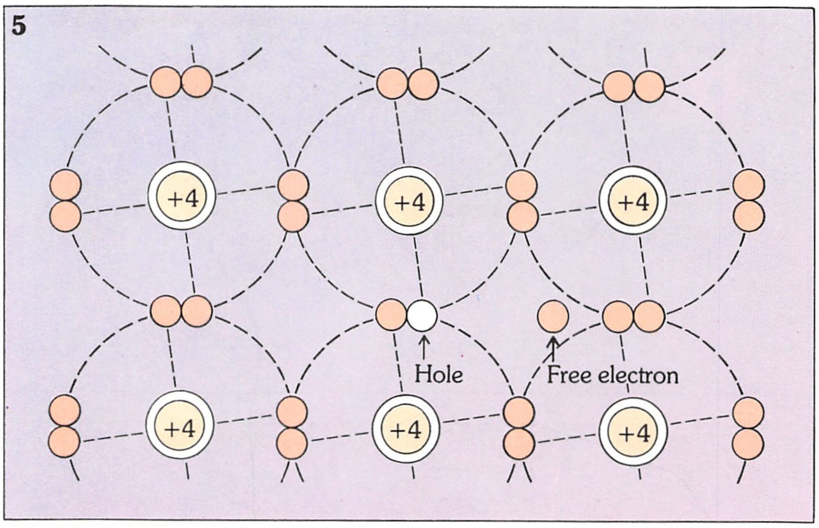
What happens when a voltage is applied?

You will remember that a current is defined as the orderly flow of electrons in a material from the negative to the positive

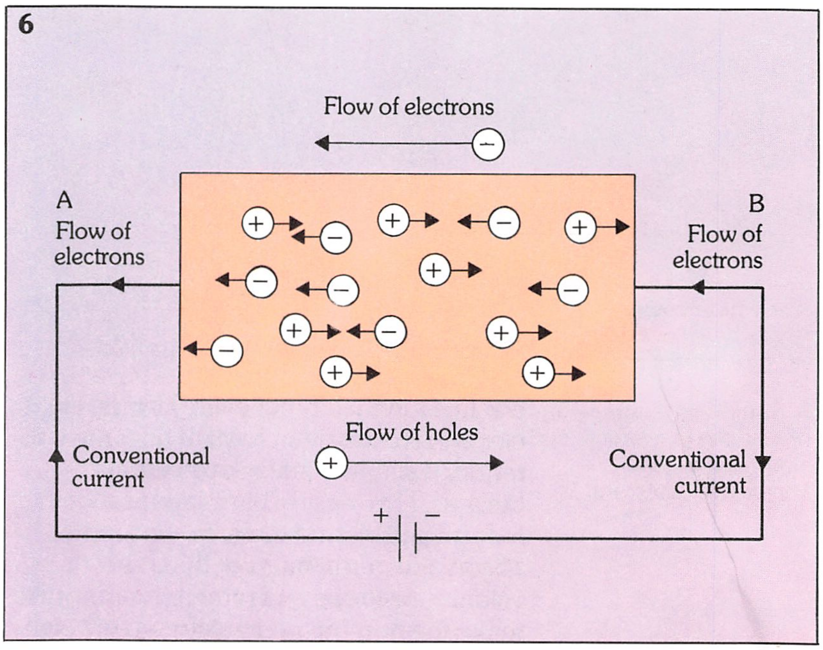
4



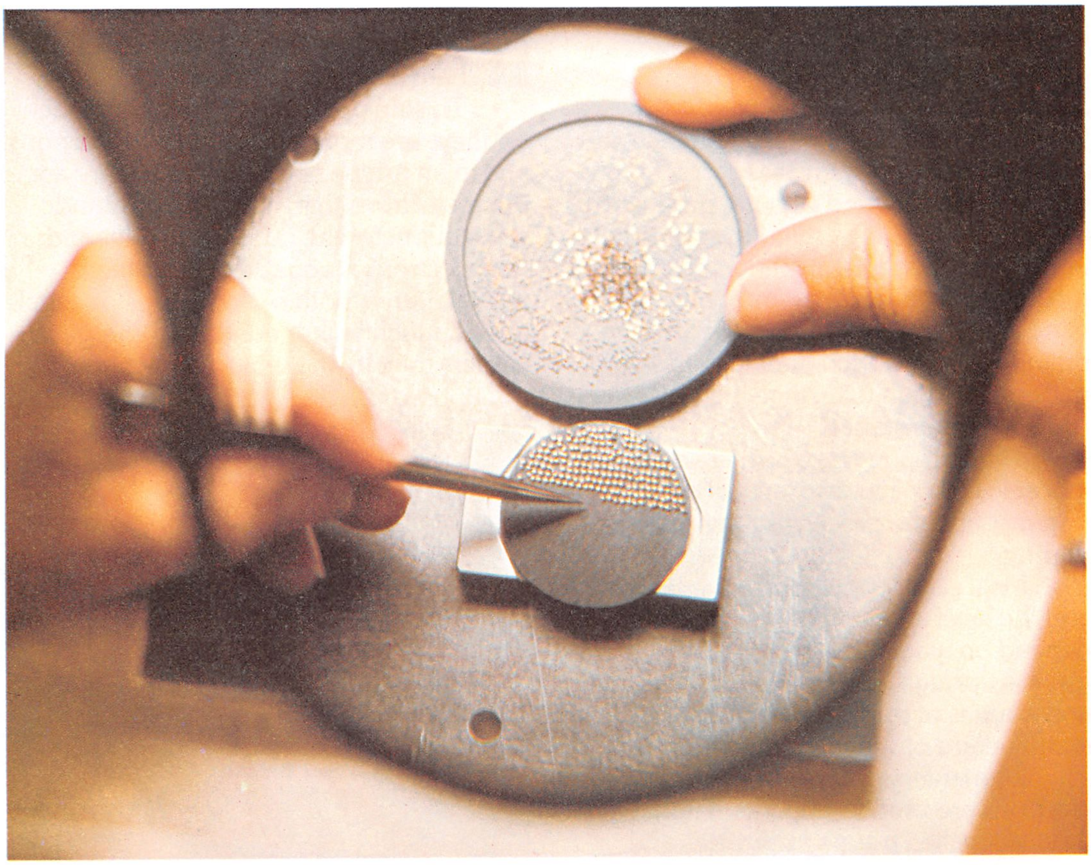
5



6



Semiconductor chips being handled during laboratory research.

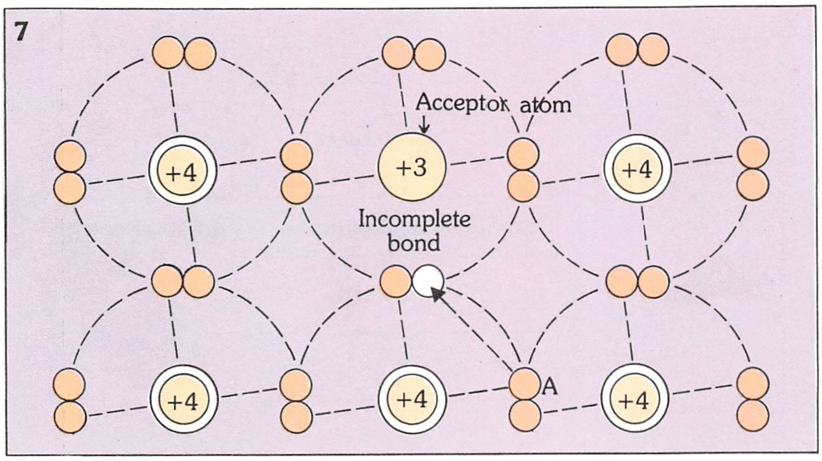


Tony Stone Photo Library – London

4. Diagrammatic cross-section of a semiconductor crystal lattice.

5. Formation of a free electron and a hole, caused by increasing the temperature of the crystal.

7. How the addition of a p-type impurity creates holes. If the electron at point A moves to fill the hole, it in turn creates a new hole at A.



6. Voltage applied to a semiconductor, showing the flow of charges and current. The symbols shown inside the semiconductor indicate the flow of electrons (–) and holes (+) which result under this applied voltage.

terminal of the applied power source.

When a voltage is applied to a semiconductor which has free electrons and holes, electrons move towards the positive terminal and holes towards the negative terminal (if electrons are going to the left, holes can be said to be moving to the right).

Figure 6 sums up the situation: the current in a semiconductor is given by the sum of two components: free electrons which move in one direction and holes which move in the other.

But it is important to remember that the movement of free holes, say to the right, is a simple way of describing the movement of **bound** electrons to the left. As each bound electron moves to fill up a hole on its left, it thereby creates a new hole on its right, in the position it previously occupied. The holes thus appear to be moving to the right. As a consequence, the movement of holes in a semiconductor will be slower than the movement of **free** electrons. In other words the mobility of holes is less than that of free electrons.

What then happens to the holes when they reach the edge of the silicon (point B in *figure 6*)? The answer is that some of the electrons coming from the negative terminal of the battery combine with the holes. The remaining electrons carry on through the silicon as free electrons. The electrons leaving at point A can be subdivided into two groups: those that entered the semiconductor at B – coming from the battery, and those that have freed themselves from atoms to become free electrons creating holes.

Extrinsic or doped semiconductors

We have mentioned that the conductivity of semiconductors increases considerably with temperature and with the presence of impurities. By adding minute amounts of impurities when a semiconductor is still in a molten state, crystals can be created with extra electrons or extra holes depending on the material added. One part in 100 million is enough to change the resistivity of silicon, for instance.

These impurities, which are added in a closely controlled way, are called **dopants** and the process is referred to as **doping.**

Most semiconductors require doping and the level and type of dopant added will depend on what the device is to be used for. It is through doping that we get the p and n-types of semiconductor used in the manufacture of semiconductor devices.

Elements with three or five valency electrons are used as dopants. Those with three are called **p-type impurities** and those with five are called **n-types.** After the impurities are added the material is known as an **extrinsic** semiconductor.

What is the significance of p- and n-types?

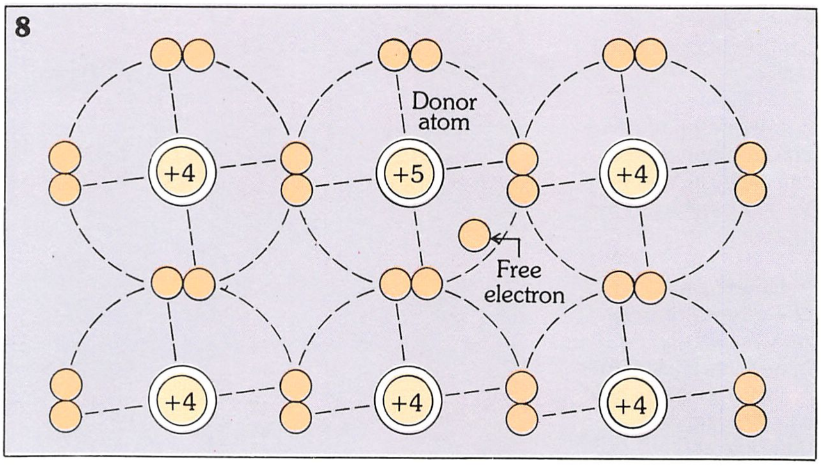
Let's first consider the p-type. The dopant atom occupies a place in the crystal structure just like the other semiconductor atoms. So what happens, for example, when an atom of boron or gallium with three valency electrons replaces an atom of silicon in the structure? With only three valency electrons one of the four covalent bonds is incomplete – that is, we have a hole (*see figure 7*). It is then easy for an electron from an adjacent atom to move into the space. So this dopant gives the semiconductor an excess of holes.

In an n-type impurity like phosphorus or arsenic there are five valency electrons. When these atoms take up their places in the crystal structure they leave a spare electron which can easily be moved from its orbit (*see figure 8*). As a consequence, n-type dopants result in a semiconductor with free electrons.

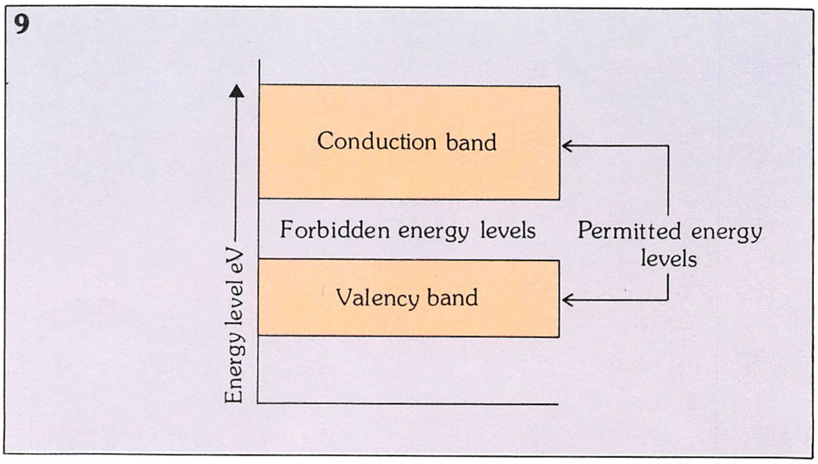
If you have difficulty remembering which is which, simply think of holes/positive charge/p-type and electrons/negative charge/n-type.

To sum up: in a p-type semiconductor there are more holes than free electrons and in an n-type semiconductor there are more electrons than holes. In an intrinsic semiconductor the number of holes and free electrons is equal. So, at ambient temperatures, both p and n-type materials have a much higher number of charge carriers than in an intrinsic semiconductor.

A slice of semiconductor material (known as a wafer), on which hundreds of integrated circuits are made.



8. The addition of n-type impurities creates free electrons.

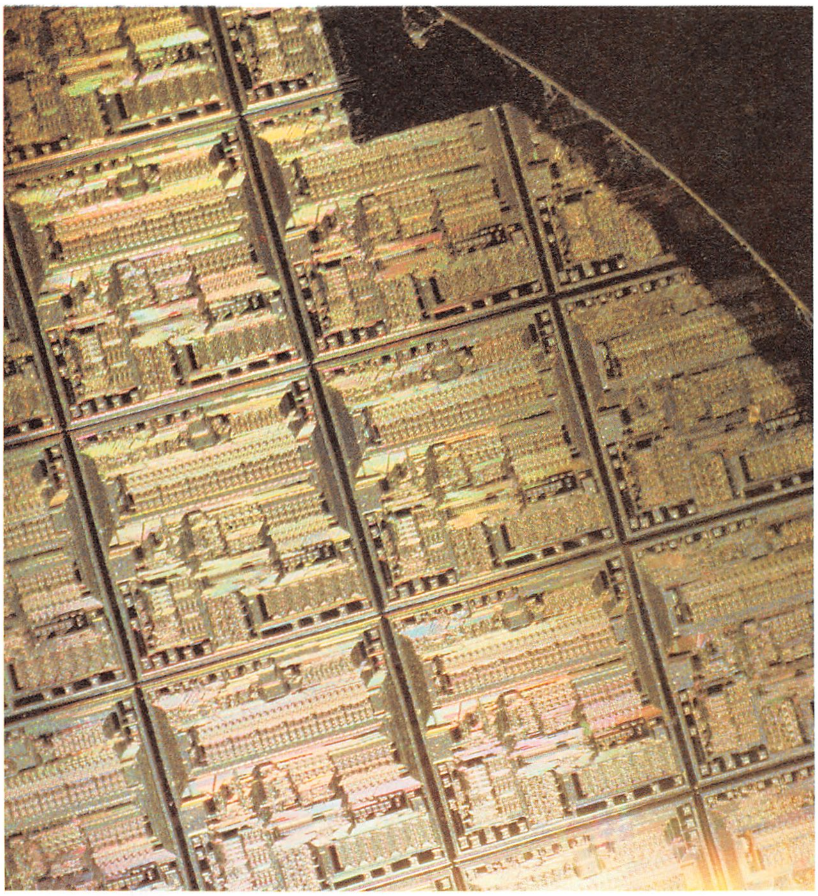


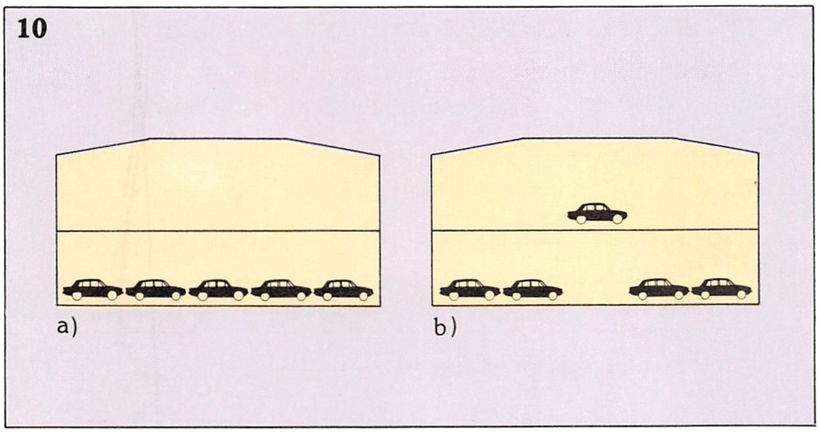
9. The permitted and forbidden energy levels for semiconductor materials.

This characteristic forms the basis of how modern semiconductor devices work.

Energy bands

There is another way of describing electrical conduction in a semiconductor. When a single atom of matter is isolated, by being placed far from any other atom, the electrons in orbit will have a certain amount of energy. This energy is distributed on different **energy levels.** The bigger the orbit, the higher the energy level





10. The car park analogy. The ground floor represents the valency band and the first floor is the conduction band. (a) represents conditions at zero temperature where no conduction is possible. (b) represents conditions as the temperature rises, and free electrons enter the conduction band.

of the electron. However, a given electron can only have certain energy levels. All the others are 'forbidden.' This means that the electron cannot be at any random energy level, but must be at a **permitted** energy level.

When on the other hand the atoms are placed very compactly, as they are in a semiconductor crystal, the permissible energy levels are modified by the presence of the adjacent atoms. These then become **permissible energy bands** – energy intervals where an electron can be placed. The remaining energy levels are forbidden and are called **prohibited bands.**

In the study of semiconductors it is important to examine the **valency** and **conduction** bands. The conduction band is found at a higher energy level than the valency band, separated by a prohibited band (see *figure 9*). In an intrinsic semiconductor at absolute zero, all the valency electrons (four per atom) are found in the valency band. As the temperature increases, some of these acquire sufficient energy to become free electrons. They cross the prohibited space and take up position in the conduction band, leaving gaps in the valency band.

The width of the prohibited energy gap indicates the amount of energy which a valency electron must gain in order to break its covalent bond and become a free electron. This energy is measured in **electronvolts** (eV). An electronvolt is the quantity of kinetic energy which an electron acquires moving between two points which have a potential difference of one volt. If this unit of measurement does not seem very significant, remember that it relates to very low overall energy levels. In silicon, the energy gap is about 1.1 eV and in germanium, about 0.7 eV. This means that less energy is required to break a bond in germanium than in silicon.

An intrinsic semiconductor at absolute zero temperature has a full valency band and an empty conduction band, so it behaves as an insulator and no electrical current can pass through it. As the temperature rises, the conduction band begins to fill up with free electrons and the valency band empties; this makes conduction possible.

You can compare this process to a car park with two floors. The ground floor corresponds to the valency band and the first to the conduction band. If the ground floor is completely full and the first one empty, as in *figure 10a,* no movement of the cars is possible at all. If a car is transferred from the ground to the first floor, movement is possible on both floors (*figure 10b*).

If n-type impurities are added to an intrinsic semiconductor, the number of free electrons in the conduction band increases considerably. If the added impurities are p-type, the number of gaps in the valency band will increase. In n-type semiconductors, the greater part of the current is made up of free electrons in the conduction band. These are called **majority carriers.** The gaps are called **minority carriers.** In p-type semiconductors the situation is reversed: the greater part of the current is caused by holes in the valency band and these are the majority carriers, while the free electrons are the minority carriers.

Glossary

acceptor impurities atoms which have fewer valency electrons than are needed to complete bonds with neighbouring atoms, and which accept electrons to complete the bonds

carrier charge carrier – a mobile electron or hole which carries charge in a semiconductor

conduction band high energy level in which electrons move and conduct in a semiconductor

donor impurities atoms which have more valency electrons than are needed to complete bonds with neighbouring atoms. They can give up their electrons to the conduction band very easily

dopant donor or acceptor impurity which is added to a pure (intrinsic) semiconductor to form either n or p-type (extrinsic) semiconductor material

extrinsic semiconductor doped semiconductor, either n-type or p-type, in which impurities determine the charge carrier concentration

hole an empty energy level in the valency band of a semiconductor, which is due to an electron being lost because of heating or being trapped by an acceptor impurity

intrinsic semiconductor pure semiconductor that has an equal number of holes and electrons

n-type semiconductor extrinsic semiconductor which has an excess of carrier electrons

p-type semiconductor extrinsic semiconductor which has an excess of carrier holes

prohibited band energy level between the conduction and valency bands which electrons can only cross to get from one state to the other

recombination process in which liberated electrons recombine with holes and restore thermal equilibrium in a semiconductor

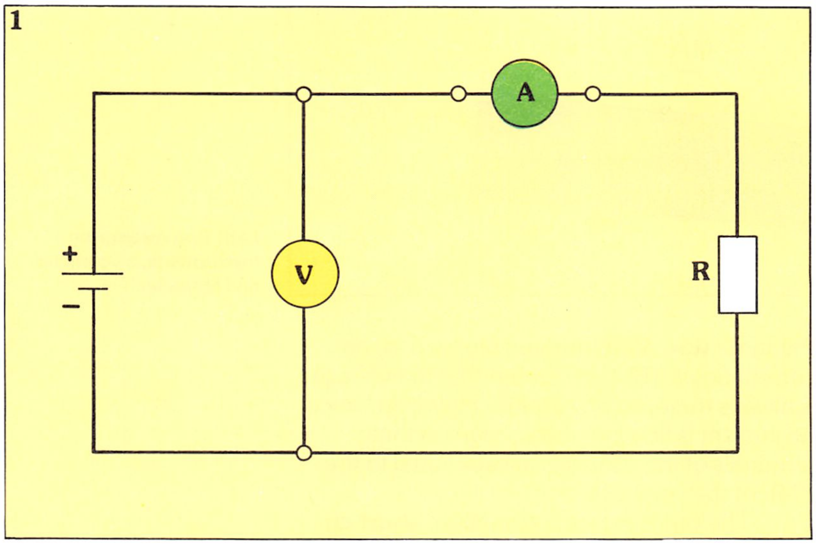
valency band low energy level in which at absolute zero (0°K) all the valency electrons are found and conduction cannot occur

valency electrons electrons which occupy the outermost energy levels of an atom and which take part in the formation of valency bonds

Understanding Ohm's law

The fundamental relationship between current, voltage and resistance forms the basis of all electrical circuits. This relationship is known as **Ohm's law.**

By looking at some measurements made on an electrical circuit using an ammeter and a voltmeter (as shown in *figure 1*), it is possible to get a better understanding of this relationship. The ammeter is connected in such a way that all the circuit current flows through it and the voltmeter is connected so that it measures all the voltage applied to the circuit.



1. Connection of an ammeter and voltmeter in an electrical circuit.

Table 1 First set of measurements with R constant = 450 Ω

Voltage (V)	4.5	9	13.5	18
Current (mA)	10	20	30	40

Table 2 Second set of measurements with V constant at 18 V

Resistance (Ω)	450	2 × 450 = 900	4 × 450 = 1800
Current (mA)	40	20 = ½ × 40	10 = ¼ × 40

The first set of measurements are made under the following conditions:

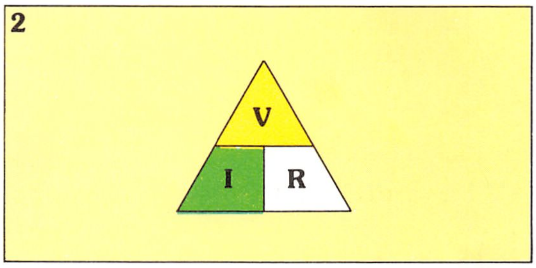
– the resistance (R) has a value of 450 ohms and will not vary significantly because it is made with constantan wire (an alloy of copper and nickel).

– the voltage is progressively increased.

The results which show current intensity against voltage level are given in *table 1*.

From this it can be seen that for a fixed resistance value, the current intensity (I) increases in direct proportion to the increase in voltage (V). In other words, in a circuit with a fixed resistance the current is directly proportional to the voltage.

A different series of measurements highlights another concept. The circuit is still the same but this time the voltage will be kept constant and the resistance varied. The results for the second series of measurements are shown in *table 2*. This time you can see that as the resistance is increased the current decreases. In other words, in a circuit with a fixed voltage the current is inversely proportional to the resistance: if the resistance doubles the current is halved.



2. Diagrammatic representation of Ohm's law.

Both concepts can be stated mathematically:

$$\text{Resistance} = \frac{\text{Voltage}}{\text{Current}} \quad \text{or } R = \frac{V}{I}$$

and with regard to the units of measurement:

$$1\Omega = \frac{1\text{V}}{1\text{A}} = 1 \times \frac{\text{V}}{\text{A}}$$

This is Ohm's law and it can also be expressed in the following ways:

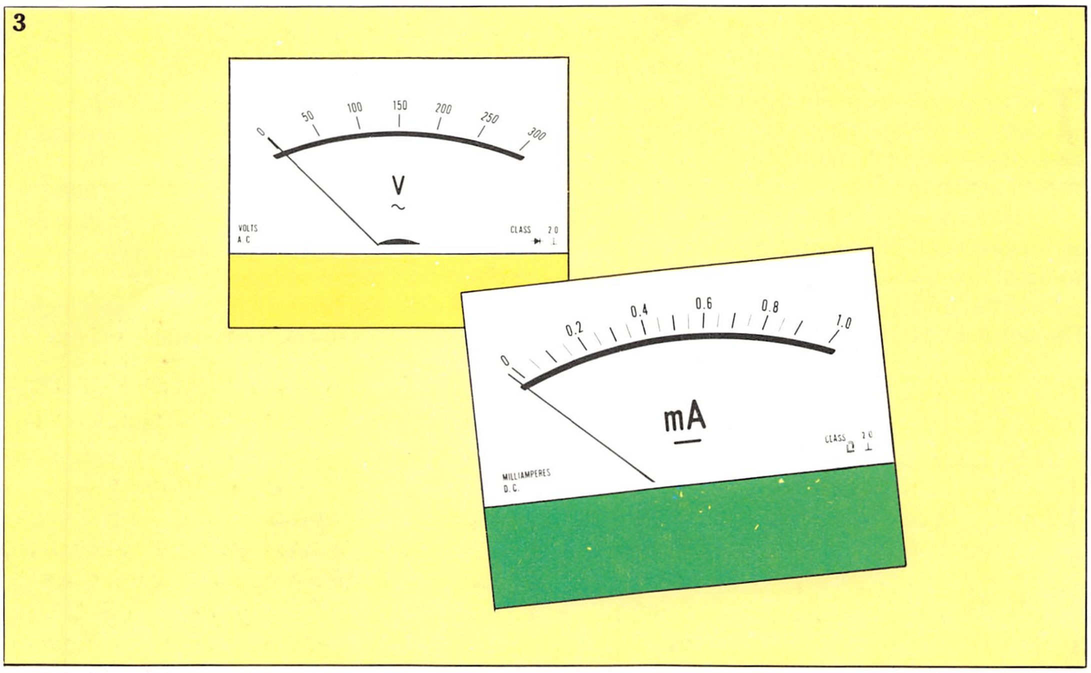
$$V = R \times I \qquad \text{or } I = \frac{V}{R}$$

Therefore any one of the three fundamental circuit values can be found if the other two are known.

Figure 2 shows a diagrammatic representation of the three values, which you may find helps you to remember the relationship between them.

Here are three examples to help clarify what has been covered so far.

1) A user knows his circuit load resistance should be 880 ohms. He wants to calculate the current (I) when a voltage of 220 V is applied to the circuit. Using *figure 2* you simply cover the value to be found (I), and you are left with the formula to find it (V/R). The calculation then is:



3

Left: Two measuring instruments, a voltmeter and ammeter.

$$I = \frac{V}{R} = \frac{220\ V}{880\ \Omega} = 0.25\ A$$

When the circuit is measured with an ammeter it should closely correspond to this value.

2) In a circuit with a resistance of 400 ohms, what voltage must be applied to give a current of 0.25 A? In this case, cover the V in *figure 2* and you are left with I/R. Therefore the formula to use as a basis for your voltage calculation is:

$$V = R \times I = 400 \times 0.25 = 100\ V$$

3) A load absorbs a current of 0.3 A with a voltage of 27 V. To find the formula to calculate the load resistance, cover up the R in *figure 2*. This time you are left with V/I thus:

$$R = \frac{V}{I} = \frac{27\ V}{0.3\ A} = 90\ \Omega$$

When we described the power supply previously it was said that some voltage was lost in pushing the current through the internal resistance. Now it can be seen that the voltage loss is due to an Ohm's law effect.

An electric circuit has two extreme conditions: open circuit and short circuit.

An **open circuit** exists when the circuit includes an infinitely high resistance: this is normally when the switch is open (i.e. in its off position). In this condition no current will flow. Think of a generator (a dynamo, for example) that keeps on turning but doesn't supply any current if the circuit switch is open. Even the wall sockets in your home can be thought of in the same way. With nothing plugged in, no current flows. (But remember that the voltage is always there, so be careful.) However, since no current is flowing, the voltage on the terminals of the source is always equal to the EMF of the source: V = E.

The other extreme condition, **short circuit**, occurs when the circuit has an almost infinitely small resistance. For example, when the terminals of a car battery are connected directly together or when the lead of an electrical appliance is cut and the copper wires touch each other. The short circuit current when associated with a sufficiently high voltage, can be very dangerous. The current that flows through the circuit is limited only by the internal resistance R_i of the source and is therefore very high:

$$\text{Short circuit current} = \frac{E}{R_i}$$

In practice, fuses or other types of security devices guard against this by becoming open circuits when too much current flows.

The human body can also be considered as a 'load' in an electrical circuit. It has an electrical resistance that can vary both from person to person and, in any person, from moment to moment. The skin has the highest resistance while the internal organs have a very low resistance. The skin's ability to withstand electric shocks varies from a few hundred ohms to many thousands of ohms depending on state of health, humidity and anxiety. Old fashioned lie-detector machines used this fact as the basis of their operation. □

3

BASIC COMPUTER SCIENCE

Data and instruction codes

How a computer works with numbers

For a real understanding of how computers work, it's important to go back to basics and see how the numerical codes they use are constructed. A computer program is essentially a coded message, which is translated by the computer into its **machine code**, made up of binary digits.

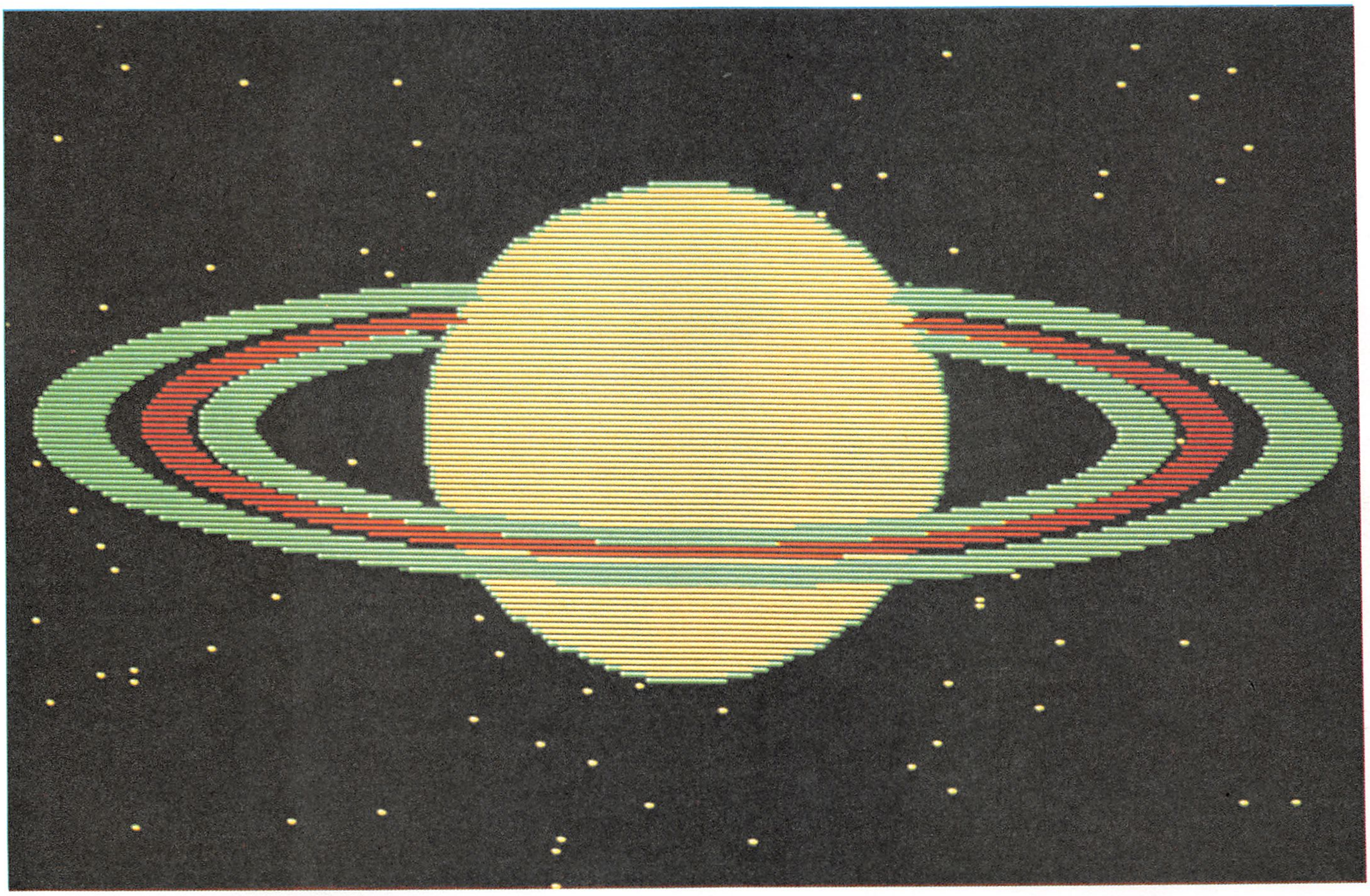
We'll look at how different binary combinations can be used to represent all the information that a computer needs to operate a program – i.e. numerical and alphabetical data, and the machine instructions which control program operation.

The arithmetic of computers

In order to do any work, the computer must be able to represent and store the numerical or alphabetical data it has to work on, and the instructions which tell it what to do.

Let's look first at how numerical information is represented.

We are used to representing numbers by means of the decimal numbering system. It consists of ten symbols (0, 1, 2, . . . 9), which allow any quantity to be represented by using a **positional criterion.** That is to say, each figure in a number assumes a different value according to where it is written in relation to others. A one written to the left of a nine indicates



A digitally created video picture of Saturn. Even complex computer graphics ultimately rely on digital codes based on 0 and 1. (photo IBM).

the number nineteen – one ten and nine ones.

In every number system we have to know how to define the value and the position of each digit in a number. Digits are **weighted** according to their position. In the decimal system numbers are weighted according to a power of ten, as this is the **base** number. For example the number five hundred can be thought of as 5×10^2. The figure 5 is known as the **coefficient,** and the 10^2 is the **weight.** As another example, five thousand can be thought of as:

Coefficient	×	Weight	= Product
5	×	10^3	= 5000

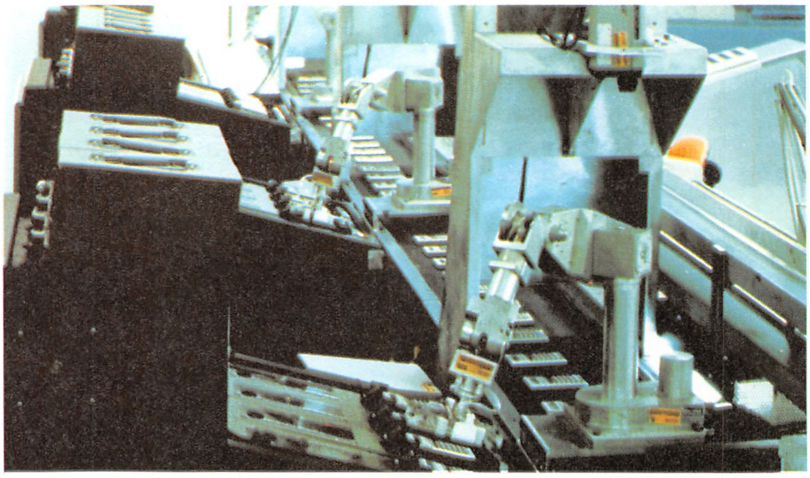
The figures of the number 567 have the following values:

Coefficient	×	Weight	= Product
5	×	$100(10^2)$	= 500
6	×	$10(10^1)$	= 60
7	×	$1(10^0)$	= 7 +
			= 567

The weights here are 10^0, 10^1, 10^2. As you expect the value of the coefficient can vary from 0 to 9. The first position is called units, the second tens, the third hundreds and so on. If there is a 9 in the unit position and 1 is added, the figure in that position becomes zero and the next position to the left is increased by one (carried). Moving a figure to the next position to the left is the same as giving it a value ten times greater. In short, each whole decimal number can be expressed as the sum of the product of the individual digit and its respective weight (which is ten to the according power). The decimal system of counting is known as 'to the base 10'.

The binary system of counting is based on two symbols, 1 and 0. It is ideally suited for use by computers as they are made up of components which are capable of presenting two opposing distinct states. For example, a magnetic tape can be magnetised in either of two directions; a transistor, like a switch, can be on or off. The binary system can be used to represent both numbers and letters according to the way in which it is used.

Although programming languages enable users to converse with computers in ordinary characters and figures we need to understand the binary system to know how a computer works.



Numerical-control robots being used to assemble computers.

What the binary system is

Remember, the binary system is arranged according to the base number two and represents numerical quantities using rules which are similar to the decimal system. The difference is that it only uses the two symbols, 0 and 1. The binary system is also a positional numerical system. The weight of each single digit within a number is a power of 2, instead of a power of 10 as in the decimal system. Brackets and the base written at the bottom right of the figure are used to avoid confusion between numbers in different bases.

The binary number 1011 (reading as 'one, zero, one, one', *not* 'one thousand and eleven') can be converted into the decimal equivalent as follows:

Coefficient	×	Weight	= Product_{10}
1	×	2^3	= 8
0	×	2^2	= 0
1	×	2^1	= 2
1	×	2^0	= 1
Decimal total			= 11

Converting decimal to binary numbers

To convert a decimal number into binary the **'successive division'** method is used. Suppose that you wish to find the binary representation of the number 1983. Divide it by two and you will get a whole number and a remainder which is either 0 or 1.

Now divide the result by 2 and repeat the procedure until you get the answer 0. The remainders of the successive divisions, taken in order from the first to the last give the binary representation of the number from the least to the most significant digits. The **least significant digit** is the one which corresponds to power zero of the base and is the one we write on the far right of a number. The **most significant digit** corresponds to the greatest power of the base.

Table 1: Adding and multiplying in binary. In computer notation, multiplication is shown by an asterisk to prevent confusion with the letter X.

Table 1

Addition and multiplication in the binary system

+	0	1
0	0	1
1	1	10

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0 with 1 to carry

×	0	1
0	0	0
1	0	1

0 × 0 = 0
0 × 1 = 0
1 × 0 = 0
1 × 1 = 1

1983	÷	2	=	991 remainder 1
991	÷	2	=	495 remainder 1
495	÷	2	=	247 remainder 1
247	÷	2	=	123 remainder 1
123	÷	2	=	61 remainder 1
61	÷	2	=	30 remainder 1
30	÷	2	=	15 remainder 0
15	÷	2	=	7 remainder 1
7	÷	2	=	3 remainder 1
3	÷	2	=	1 remainder 1
1	÷	2	=	0 remainder 1

which gives us the binary number (11110111111).

The rules for carrying out arithmetic in the binary system are the same as decimal ones. See *table 1.*

This is how whole numbers can be converted from the binary system to decimal and vice versa (we'll look at fractions later on). These conversion operations, although simple in concept, are long and boring to carry out, and there are other disadvantages – even small numbers are lengthy to write and there is a high possibility of error in manipulating so many digits.

How numbers can be simplified

There are two number systems which can help lessen the difficulties caused by binary; they are the **octal** and **hexadecimal** systems. These are both more compact and can be easily converted back into decimal or binary.

The octal number system is arranged according to the base 8. It is a positional system and uses the eight symbols 0 to 7. As an example let's convert the octal number 536 into decimal:

Coefficient	×	Weight	= Product$_{10}$
5	×	8^2	= 320
3	×	8^1	= 24
6	×	8^0	= 6
Decimal total			= 350

Example of an on-line computer facility. The terminals are connected to a central computer.

Tony Stone Photo Library – London

Decimal numbers can be converted into octal by use of the successive division method. For instance to reconvert the decimal number 350 into octal:

$350 \div 8 = 43$ remainder 6
$43 \div 8 = 5$ remainder 3
$5 \div 8 = 0$ remainder 5

Remember that the last remainder of this operation is the most significant digit of the converted number. So the answer is: $(350)_{10} = (536)_8$.
The results of octal multiplication and addition are summarized in *table 2*.

The conversion of an octal number into binary and vice versa can be done very quickly. Bearing in mind that $8 = 2^3$ you can see that a binary number can be converted into octal by grouping its digits in threes, from the least to the most significant, and then converting each binary group into the corresponding octal digit by reading each group as a separate binary number. For instance, suppose we convert the binary number 1001001101101110 into octal:

Binary	1	001	001	101	101	110
Octal	1	1	1	5	5	6

An octal number can be easily converted into binary by substituting the corresponding three digit binary number for each octal digit.

Octal	1	1	1	5	5	6
Binary	1	001	001	101	101	110

This is much quicker than using the successive division method and in fact the octal system is often used to represent the internal information of a computer in a more compact and manageable form.

The hexadecimal number system
The hexadecimal system is arranged according to the base 16. The numbers 0 to 9 are represented by the conventional symbols but 10 to 15 are represented by the letters A to F. The usual rules of number conversion apply. For instance let's find the decimal values of the hexadecimal number A3B:

Coefficient	×	Weight	= Product_{10}
A	×	16^2	= 2560
3	×	16^1	= 48
B	×	16^0	= 11
Decimal total			= 2619

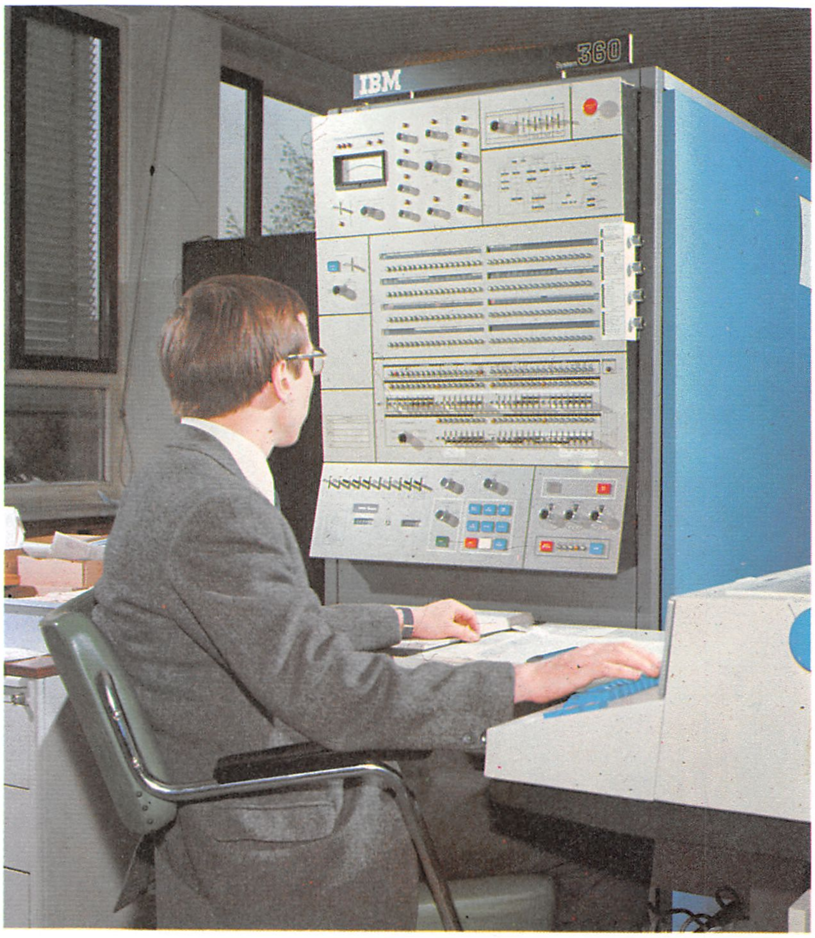
Or to turn 2619 back into hexadecimal we use successive division:

$2619 \div 16 = 163$ remainder $11 = B$
$163 \div 16 = 10$ remainder $3 = 3$
$10 \div 16 = 0$ remainder $10 = A$

Table 2
Addition and multiplication in the octal system

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	10
2	2	3	4	5	6	7	10	11
3	3	4	5	6	7	10	11	12
4	4	5	6	7	10	11	12	13
5	5	6	7	10	11	12	13	14
6	6	7	10	11	12	13	14	15
7	7	10	11	12	13	14	15	16

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	10	12	14	16
3	0	3	6	11	14	17	22	25
4	0	4	10	14	20	24	30	34
5	0	5	12	17	24	31	36	43
6	0	6	14	22	30	36	44	52
7	0	7	16	25	34	43	52	61



ZEFA

The main operating console of a large computer system. The operator can check and control any piece of hardware being used, as well as enter and run programs manually.

Table 3

Addition and multiplication in the hexadecimal system

+	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
1	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11
2	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12
3	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
4	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14
5	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15
6	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16
7	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17
8	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18
9	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19
A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A
B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B
C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E
F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20

×	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	04	06	08	0A	0C	0E	10	12	14	16	18	1A	1C	1E	20
3	06	09	0C	0F	12	15	18	1B	1E	21	24	27	2A	2D	30
4	08	0C	10	14	18	1C	20	24	28	2C	30	34	38	3C	40
5	0A	0F	14	19	1E	23	28	2D	32	37	3C	41	46	4B	50
6	0C	12	18	1E	24	2A	30	36	3C	42	48	4E	54	5A	60
7	0E	15	1C	23	2A	31	38	3F	46	4D	54	5B	62	69	70
8	10	18	20	28	30	38	40	48	50	58	60	68	70	78	80
9	12	1B	24	2D	36	3F	48	51	5A	63	6C	75	7E	87	90
A	14	1E	28	32	3C	46	50	5A	64	6E	78	82	8C	96	A0
B	16	21	2C	37	42	4D	58	63	6E	79	84	8F	9A	A5	B0
C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4	C0
D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3	D0
E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2	E0
F	1E	2D	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1	F0
10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0	100

which gives us the hexadecimal number $(A3B)_{16}$.

As for the octal system, the conversion of hexadecimal numbers into binary and vice versa can be done in one operation. However since $16 = 2^4$, the groups are of four binary digits and not three. For instance to convert the binary number 1001001101101110 into hexadecimal:

Binary	1001	0011	0110	1110
Decimal	9	3	6	14
Hexadecimal	9	3	6	E

Or to reconvert the hexadecimal number 936E into binary:

Hexadecimal	9	3	6	E
Binary	1001	0011	0110	1110

Table 3 shows the result of hexadecimal addition and multiplication. *Table 4* compares the first 17 numbers of the decimal binary and hexadecimal systems.

How to convert fractions into different number systems

Now let's look at the conversion of positive fractions to and from number systems with different bases. In a decimal number less than one the weights of the numbers to the right of the decimal point correspond to the negative powers of ten from −1 onwards. For example:

$$\begin{aligned} 0.6483 = \quad & 6 \times 10^{-1} + \\ & 4 \times 10^{-2} + \\ & 8 \times 10^{-3} + \\ & 3 \times 10^{-4} \end{aligned}$$

The same principle is true for number

systems with bases other than 10. So the conversion of a binary, octal or hexadecimal equivalent is as follows:

$$\begin{array}{llr}
0.11001_2 = & 1 \times 2^{-1} & = 0.5 \\
 & 1 \times 2^{-2} & = 0.25 + \\
 & 0 \times 2^{-3} & = 0.00 + \\
 & 0 \times 2^{-4} & = 0.00 + \\
 & 1 \times 2^{-5} & = 0.03125 + \\
 & & = 0.78125
\end{array}$$

Converting a decimal fraction into a different number system is done using the **successive multiplication** method, as follows.

The original (decimal) number is multiplied by the base of the new number system. The whole number part of the answer gives the first figure to the right of the decimal point in the new representation. The answer, with the whole number removed, is again multiplied by the base. The whole number part of this answer is the second figure to the right of the decimal point. This procedure continues until a complete answer is obtained.

For example let's convert the decimal fraction $(0.90625)_{10}$ into binary, octal and hexadecimal.

$$\begin{array}{llllc}
0.90625 & \times & 2 & = 1.81250 & 1 \\
0.81250 & \times & 2 & = 1.6250 & 1 \\
0.6250 & \times & 2 & = 1.250 & 1 \\
0.25 & \times & 2 & = 0.5 & 0 \\
0.5 & \times & 2 & = 1.0 & 1
\end{array}$$

which gives us the binary fraction 0.11101

$$\begin{array}{llllc}
0.90625 & \times & 8 & = 7.25 & 7 \\
0.25 & \times & 8 & = 2.00 & 2
\end{array}$$

which gives us the octal fraction 0.72

$$\begin{array}{lllll}
0.90625 & \times & 16 & = 14.5 & 14(\mathrm{E}) \\
0.5 & \times & 16 & = \ 8.0 & 8
\end{array}$$

which gives us the hexadecimal fraction 0.E8.

As with whole numbers, conversion from binary to octal to hexadecimal etc is easily done. To change a binary fraction to its octal equivalent, just group the digits on the right of the point in threes from left to right. The last must have sufficient zeros added to make a group of three. Then give each group its corresponding octal digit.

Table 4
Comparison of numbers to different bases

Base 10 Decimal	Base 2 Binary	Base 16 Hexadecimal
0	00000	0
1	00001	1
2	00010	2
3	00011	3
4	00100	4
5	00101	5
6	00110	6
7	00111	7
8	01000	8
9	01001	9
10	01010	A
11	01011	B
12	01100	C
13	01101	D
14	01110	E
15	01111	F
16	10000	10

$$\begin{array}{l}
\ \ 0.\ 11101_2 \\
= 0.\ 111\ 010_2 \\
= 0.\ \ 7\ \ \ \ 2_8
\end{array}$$

To convert an octal fraction to binary just replace each digit with the corresponding group of three bits. With binary-hexadecimal conversion the same theory applies, only this time the groups are of four bits.

$$\begin{array}{l}
\ \ 0\,.\ 11101_2 \\
= 0\,.\ 1110\ 1000_2 \\
= 0\,.\ \ \ \mathrm{E}\ \ \ \ \ 8_{16}
\end{array}$$

Occasionally when converting a fraction you will get a **periodic number**, that's to say one that has an infinite number of fractional digits which go on repeating periodically. For instance if we convert $(0.1)_{10}$ into binary:

$$\begin{array}{llllc}
0.1 & \times & 2 & = 0.2 & 0 \\
0.2 & \times & 2 & = 0.4 & 0 \\
0.4 & \times & 2 & = 0.8 & 0 \\
0.8 & \times & 2 & = 1.6 & 1 \\
0.6 & \times & 2 & = 1.2 & 1 \\
0.2 & \times & 2 & = 0.4 & 0 \\
0.4 & \times & 2 & = 0.8 & 0
\end{array}$$

and so on.

It is impossible to get an exact binary equivalent of a periodic number so they are approximated by computers.

Typical home computer keyboard for the input of instructions in the simple programming language BASIC. The computer translates these instructions into a binary-based machine code it can understand, to carry out the required operations.

How positive and negative numbers are represented

Now let's look at how positive and negative numbers are distinguished by the computer. To simplify this we shall take the case of a four bit memory word. The different configurations of bits which can be represented in this are:

0000	0100	1000	1100
0001	0101	1001	1101
0010	0110	1010	1110
0011	0111	1011	1111

These are the positive whole numbers from 0 to 15, that is to say from 0 to 2^4-1. To generalise, we can say that in a word N bits long, all the positive whole numbers from 0 to 2^N-1 can be represented. To represent both positive and negative numbers we can use half the configurations for positive numbers and the other half for the negative numbers.

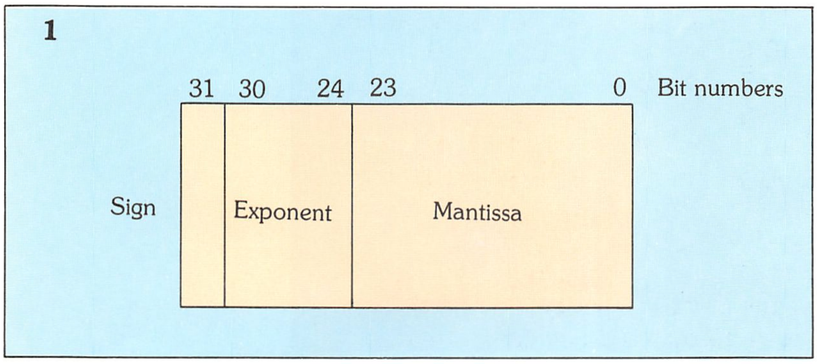
To go back to the 4 bit word, the first 8 configurations from 0000 to 0111 are used to represent positive numbers from 0 to +7, the second 8 configurations represent negative numbers from −8 to −1:

0000	= 0		1000	= −8	
0001	= +1		1001	= −7	
0010	= +2		1010	= −6	
0011	= +3		1011	= −5	
0100	= +4		1100	= −4	
0101	= +5		1101	= −3	
0110	= +6		1110	= −2	
0111	= +7		1111	= −1	

The representation of a negative binary number is obtained by subtracting the same *positive* number from 2_N (N = the number of bits in the word). Here the number would be subtracted from 10,000 (i.e. 2^4 expressed in binary). A simpler method is to invert the positive number bit by bit and add 1 to the result.

For example, find the representation of −5 in a 4 bit word.

1. How a single precision floating point number is represented in two 16 bit words.



+5 as a 4 bit word = 0101
Invert this = 1010
Add 1 to the result = 1011

which gives us the representation of −5 in a 4 bit binary word. This process is called **two's complement representation.** Note that the negative numbers have a 1 in the most significant bit position, while the positive numbers have a 0.

Generalizing this process, we can say that in a word of N bits we can represent positive numbers from 0 to $+(2^{N-1}-1)$ and negative numbers from -1 to -2^{N-1}.

When numbers are represented as negative, subtraction is achieved by addition (adding a negative number being the same as subtracting a positive number):

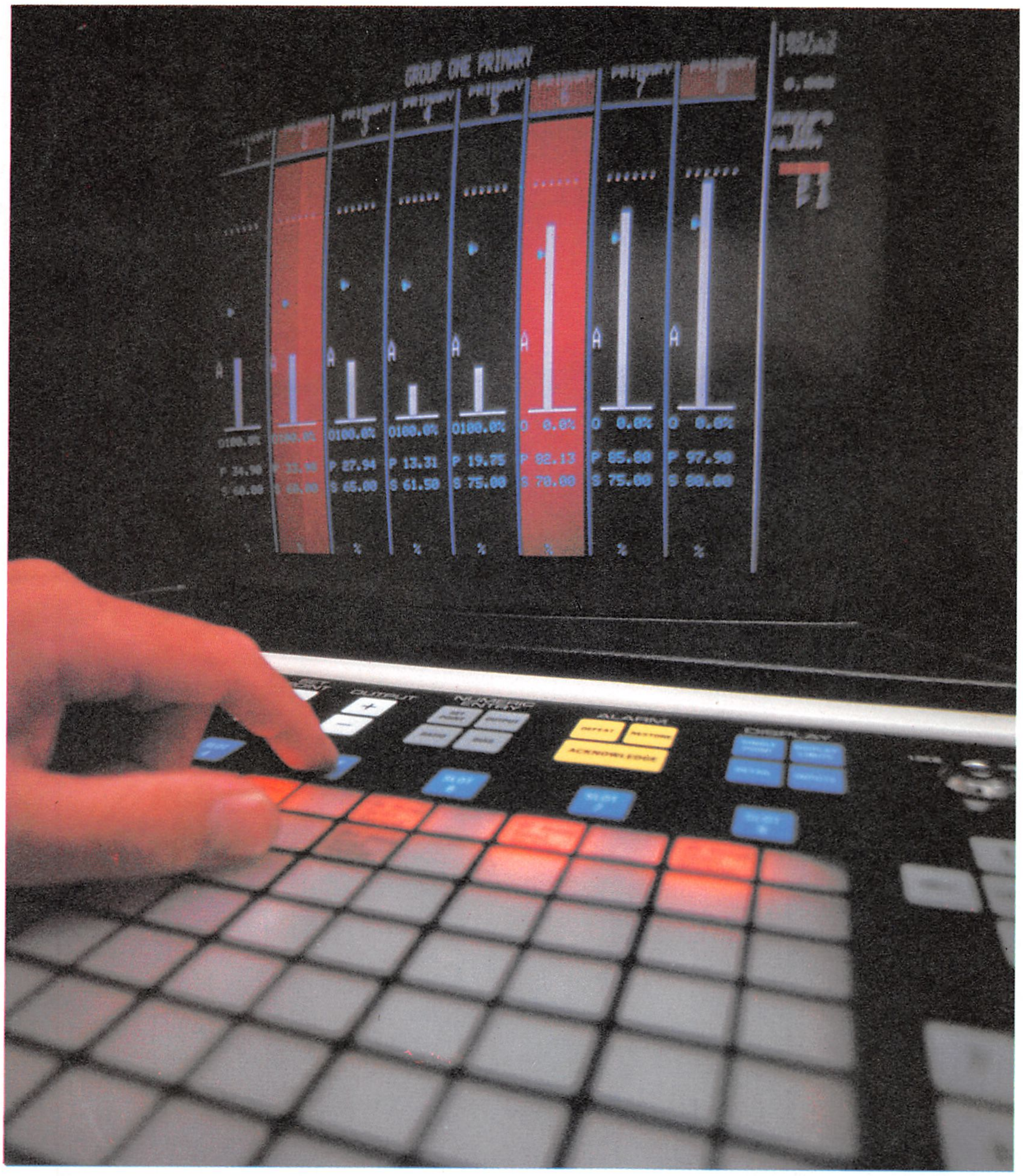
$$5 - 7 = 5 + (-7) \qquad = 0101 + 1001$$

$$\text{which is:} \quad \begin{array}{r} 0101 \\ \underline{1001} + \\ = 1110 \end{array}$$

$$1110 = -2$$

This means that both addition and subtraction can be done by the same circuits in a computer, which considerably reduces the hardware involved.

(continued in part 5)



An operating console which is in direct control of a computer system. Pressing a certain key will directly communicate a specific instruction. This type of system is used to control large industrial complexes or public services, such as power stations or oil supply terminals.

Tony Stone Photo Library – London